JC03 Rec'd PCT/PTO 03 JAN 2001

FORM PTO-1390 REV. 5-93 US DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE **TRANSMITTAL LETTER TO THE UNITED STATES DESIGNATED/ELECTED OFFICE (DO/EO/US) CONCERNING A FILING UNDER 35 U.S.C. 371**		ATTORNEYS DOCKET NUMBER **P00,1982** U.S. APPLICATION NO. (if known, see 37 CFR 1.5) **09/721000**
INTERNATIONAL APPLICATION NO. **PCT/DE99/01915**	INTERNATIONAL FILING DATE **01 JULY 1999**	PRIORITY DATE CLAIMED **03 JULY 1998**
TITLE OF INVENTION **METHOD AND ARRANGEMENT FOR DESIGNING THE CONTROL OF A COMPLETE PROCESS**		
APPLICANT(S) FOR DO/EO/US **BERTIL BRANDIN ET AL.**		

Applicant herewith submits to the United States Designated/Elected Office (DO/EO/US) the following items and other information:

1. ☒ This is a **FIRST** submission of items concerning a filing under 35 U.S.C. 371.
2. ☐ This is a **SECOND** or **SUBSEQUENT** submission of items concerning a filing under 35 U.S.C. 371.
3. ☒ This express request to begin national examination procedures (35 U.S.C. 371(f)) at any time rather than delay.
4. ☒ A proper Demand for International Preliminary Examination was made by the 19th month from the earliest claimed priority date.

5. ☒ A copy of International Application as filed (35 U.S.C. 371(c)(2)) - drawings attached.
   a. ☒ is transmitted herewith (required only if not transmitted by the International Bureau).
   b. ☐ has been transmitted by the International Bureau.
   c. ☐ is not required, as the application was filed in the United States Receiving Office (RO/US)
6. ☒ A translation of the International Application into English (35 U.S.C. 371(c)(2) - drawings attached.

7. ☒ Amendments to the claims of the International Application under PCT Article 19 (35 U.S.C. §371(c)(3))
   a. ☐ are transmitted herewith (required only if not transmitted by the International Bureau).
   b. ☐ have been transmitted by the International Bureau.
   c. ☐ have not been made; however, the time limit for making such amendments has NOT expired.
   d. ☒ have not been made and will not be made.

8. ☐ A translation of the amendments to the claims under PCT Article 19 (35 U.S.C. 371(c)(3)).

9. ☒ An oath or declaration of the inventor(s) (35 U.S.C. 371(c)(4)).

10. ☐ A translation of the annexes to the International Preliminary Examination Report under PCT Article 36 (35 U.S.C. 371(c)(5)).

Items 11. to 16. below concern other document(s) or information included:

11. ☒ An Information Disclosure Statement under 37 C.F.R. 1.97 and 1.98; **(PTO 1449, Prior Art, Search Report, 08 References).**

12. ☒ An assignment document for recording. A separate cover sheet in compliance with 37 C.F.R. 3.28 and 3.31 is included. **(SEE ATTACHED ENVELOPE)**

13. ☒ Amendment "A" Prior to Action with Appendix "A" attached.
    ☐ A SECOND or SUBSEQUENT preliminary amendment.

14. ☐ A Substitute Specification and Mark-Up for Substitute Specification.

15. ☒ A change of address letter attached to the Declaration.

16. ☒ Other items or information:
    a. ☒ Appointment of Associate Power of Attorney
    b. ☒ EXPRESS MAIL #EL655303081US dated January 3, 2001.

525 Rec'd PCT/PTO 03 JAN 2001

U.S. APPLICATION NO. (if known, see 37 C.F.R. 1.5) 09/721000	INTERNATIONAL APPLICATION NO. PCT/DE99/01915	ATTORNEY'S DOCKET NUMBER P00,1982

17. ☒ The following fees are submitted:	CALCULATIONS	PTO USE ONLY
BASIC NATIONAL FEE (37 C.F.R. 1.492(a)(1)-(5): Search Report has been prepared by the EPO or JPO .... $860.00 International preliminary examination fee paid to USPTO (37 C.F.R. 1.482) .. $690.00 No international preliminary examination fee paid to USPTO (37 C.F.R. 1.482) but international search fee paid to USPTO (37 C.F.R. 1.445(a)(2) .... $710.00 Neither international preliminary examination fee (37 C.F.R. 1.482) nor international search fee (37 C.F.R. 1.445(a)(2) paid to USPTO .... $1000.00 International preliminary examination fee paid to USPTO (37 C.F.R. 1.482) and all claims satisfied provisions of PCT Article 33(2)-(4) .... $ 100.00		
ENTER APPROPRIATE BASIC FEE AMOUNT =	$ 860.00	
Surcharge of $130.00 for furnishing the oath or declaration later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 C.F.R. 1.492(e)).	$	

Claims	Number Filed		Number Extra	Rate		
Total Claims	08	- 20 =	0	X $ 18.00	$	
Independent Claims	02	- 3 =	0	X $ 80.00	$	
Multiple Dependent Claims				$270.00+	$	
TOTAL OF ABOVE CALCULATIONS =					$ 860.00	
Reduction by ½ for filing by small entity, if applicable. Verified Small Entity statement must also be filed. (Note 37 C.F.R. 1.9, 1.27, 1.28)					$	
SUBTOTAL =					$ 860.00	
Processing fee of $130.00 for furnishing the English translation later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492(f)). +					$	
TOTAL NATIONAL FEE =					$ 860.00	
Fee for recording the enclosed assignment (37 C.F.R. 1.21(h). The assignment must be accompanied by an appropriate cover sheet (37 C.F.R. 3.28, 3.31). $40.00 per property +						
TOTAL FEES ENCLOSED =					$ 860.00	
					Amount to be refunded	$
					charged	$

a. ☒ A check in the amount of $ 860.00 to cover the above fees is enclosed.

b. ☐ Please charge my Deposit Account No. ____________ in the amount of $ ________ to cover the above fees. A duplicate copy of this sheet is enclosed.

c. ☒ The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. 50-1519. A duplicate copy of this sheet is enclosed.

NOTE: Where an appropriate time limit under 37 C.F.R. 1.494 or 1.495 has not been met, a petition to revive (37 C.F.R. 1.137(a) or (b)) must be filed and granted to restore the application to pending status.

SEND ALL CORRESPONDENCE TO:

SCHIFF HARDIN & WAITE
PATENT DEPARTMENT
6600 Sears Tower
233 South Wacker Drive
Chicago, Illinois 60606-6473

Mark Bergner
SIGNATURE

Mark Bergner
NAME

45,877
Registration Number

09/721000
525 Rec'd PCT/PTO 03 JAN 2001

BOX PCT
IN THE UNITED STATES DESIGNATED/ELECTED OFFICE
OF THE UNITED STATES PATENT AND TRADEMARK OFFICE
UNDER THE PATENT COOPERATION TREATY--CHAPTER II

APPLICANT(S): BERTIL BRANDIN ET AL
ATTORNEY DOCKET NO.: P00,1982
INTERNATIONAL APPLICATION NO: PCT/DE99/01915
INTERNATIONAL FILING DATE: 01 JULY 1999
INVENTION: METHOD AND ARRANGEMENT FOR DESIGNING THE CONTROL OF A COMPLETE PROCESS

Assistant Commissioner for Patents,
Washington D.C. 20231

AMENDMENT A PRIOR TO ACTION

Sir:

Applicants herewith amend the above-referenced PCT application, and request entry of the Amendment prior to examination on the United States Examination Phase.

IN THE CLAIMS:

On page 13:

replace line 1 with --WHAT IS CLAIMED IS:--;

Please replace original claims 1-8 with the following rewritten claims 1-8, referring to the mark-ups in Appendix A.

1. (Amended) A method for designing a control of a complete process which comprises a number of individual processes, said method comprising the steps of:

a) identifying functionalities of said individual processes;

b) performing a validation by automatically verifying an interplay of said functionalities in accordance with an input to said complete process, while not impeding each individual process during an operation, producing a validation result; and

c) determining data for controlling said complete process from said validation result.

2. (Amended) The method as claimed in claim 1, further comprising the step of performing a sequence optimization.

3. (Amended) The method as claimed in claim 1, further comprising the step of producing data for said control in an executable code form.

4. (Amended) The method as claimed in claim 1, further comprising the step of controlling individual affected processes by a software unit which is one of said functionalities of said individual processes.

5. (Amended) The method as claimed in claim 1, wherein one or more of said individual processes may be an impeding process, an impeding process being defined as such if one of the following conditions is met:

a) an individual process is blocked by another individual process; and

b) an individual process reaches an unauthorized state or a state endangering operation of said complete system.

6. (Amended) The method as claimed in claim 1, further comprising the steps of:

designing an automatic placement machine; and

controlling individual processes of said machine.

7. (Amended) The method as claimed in claim 1, further comprising the step of controlling a technical installation with data determined for controlling said complete process.

8. (Amended) An arrangement for designing the control of a complete process, comprising:

a number of individual processes; and

a processor unit configured to provide:

a) identification of functionalities of said individual processes;

b) a validation, by automatically verifying an interplay of functionalities in accordance with an input to said complete process, in a manner such that each of said individual processes is not impeded during an operation; and

c) data from a result of said validation that is used for controlling said complete process.

REMARKS

The present Amendment revises the specification and claims to conform to United States patent practice, before examination of the present PCT application in the United States National Examination Phase. Pursuant to 37 CFR 1.125 (b), applicants have concurrently submitted a substitute specification, excluding the claims, and provided a marked-up copy. All of the changes are editorial and applicant believes no new matter is added thereby. The amendment, addition, and/or cancellation of claims is not intended to be a surrender of any of the subject matter of those claims.

Early examination on the merits is respectfully requested.

Submitted by,

Mark Bergner (Reg. No. 45,877)

Mark Bergner
Schiff Hardin & Waite
Patent Department
6600 Sears Tower
233 South Wacker Drive
Chicago, Illinois 60606-6473
(312) 258-5779
Attorneys for Applicant

Appendix A
Mark Ups for Claim Amendments

This redlined draft, generated by CompareRite (TM) - The Instant Redliner, shows the differences between -
original document : Q:\DOCUMENTS\YEAR 2000\P001982-BRANDIN-\ORIGINAL CLAIMS.DOC
and revised document: Q:\DOCUMENTS\YEAR 2000\P001982-BRANDIN-\AMENDED CLAIMS.DOC

CompareRite found 55 change(s) in the text

Deletions appear as Overstrike text surrounded by []
Additions appear as Bold-Underline text

1. **(Amended)** A method for designing [~~the~~] **a** control of a complete process which comprises a number of individual processes, **said method comprising the steps of:**

a) **identifying** [~~a) in which~~] functionalities of [~~the~~] **said** individual processes [~~are identified,~~]**;**

[~~b) in which~~] **b)** **performing** a validation [~~is performed~~] by automatically verifying an interplay of [~~the~~] **said** functionalities in accordance with an input to [~~the~~] **said** complete process, [~~to the effect that~~] **while not impeding** each individual process [~~is not impeded during the operation,~~] **during an operation, producing a validation result; and**

[~~c) in which~~] **c)** **determining** data for controlling [~~the~~] **said** complete process [~~are determined from a result of the~~] **from said** validation **result**.

2. **(Amended)** The method as claimed in claim 1, [~~in which~~] **further comprising the step of performing** a sequence optimization [~~is performed in addition to step 1c).~~]**.**

[~~3.~~]**3. (Amended)** The method as claimed in claim 1 [~~or 2, in which the data for the control are determined in the form of~~]**, further comprising the step of producing data for said control in** an executable code **form.**[~~.~~]

4. **(Amended)** The method as claimed in [~~one of claims 1 to 3, in which one of the functionalities of the individual processes is a software unit for controlling the individual process affected.~~] **claim 1, further comprising the step of controlling individual affected processes by a software unit which is one of said functionalities of said individual processes.**

[~~5.~~]**5. (Amended)** The method as claimed in [~~one of the preceding claims, in which an individual process is impeded~~] **claim 1, wherein one or more of said individual processes may be an impeding process, an impeding process being defined as such** if one of the following conditions is met:

a) [~~the~~] **an** individual process is blocked by another individual process; **and**

b) [~~the~~] **an** individual process reaches an unauthorized state or a state endangering [~~the~~] operation of [~~the~~] **said** complete system.

6. **(Amended)** The method as claimed in [~~one of the preceding claims, in which the control of individual processes of~~] **claim 1, further comprising the steps of:**

designing an automatic placement machine [~~is designed.~~]**; and**

[~~7.~~]**controlling individual processes of said machine.**

7. (Amended) The method as claimed in [~~one of the preceding claims, in which the~~] **claim 1, further comprising the step of controlling a technical installation with** data determined for controlling [~~the~~] **said** complete process [~~are used for controlling a technical installation.~~]**.**

[~~8.~~]**8. (Amended)** An arrangement for designing the control of a complete process [~~which comprises~~]**, comprising:**

a number of individual processes[~~, comprising~~]**; and**

a processor unit ~~[which is set up in such a manner that]~~ **configured to provide:**

~~[a)]~~**a)** **identification of** functionalities of ~~[the]~~ **said** individual processes ~~[can be identified]~~;

b) a validation ~~[can be performed]~~, by automatically verifying an interplay of ~~[the]~~ functionalities in accordance with an input to ~~[the]~~ **said** complete process, ~~[to the effect]~~ **in a manner such** that each **of said** individual ~~[process]~~ **processes** is not impeded during ~~[the]~~ **an** operation; **and**

c) data from a result of ~~[the]~~ **said** validation ~~[can be]~~ **that is** used for controlling ~~[the]~~ **said** complete process.

09/721000

525 Rec'd PCT/PTO 03 JAN 2001

This redlined draft, generated by CompareRite (TM) - The Instant Redliner, shows the differences between -
original document : Q:\DOCUMENTS\YEAR 2000\P001982-BRANDIN-\ORIGINAL SPECIFICATION.DOC
and revised document: Q:\DOCUMENTS\YEAR 2000\P001982-BRANDIN-\SUBSTITUTE SPECIFICATION.DOC

CompareRite found 124 change(s) in the text

Deletions appear as Overstrike text surrounded by []
Additions appear as Bold-Underline text

[~~Description~~] **SPECIFICATION**

[~~Method and arrangement for designing the control of a complete process~~] **TITLE**

METHOD AND ARRANGEMENT FOR DESIGNING THE CONTROL OF A COMPLETE PROCESS

BACKGROUND OF THE INVENTION

Field of the Invention

The invention relates to a method and an arrangement for designing the control of a complete process which comprises a number of individual processes.

Description of the Related Art

The control of a complex technical installation or[~~, respectively,~~] of a system [~~(~~]**(a** complete process) comprises a number of smaller control units which are provided for certain parts (individual processes) of the installation or[~~, respectively,~~] of the system. A first control unit for a first individual process is restricted in this case to this individual process. The same applies to a second control for a second individual process. Even if an interplay of the first control with the second control functions largely without errors, this does not guarantee that an error-free operation of the complete installation is still guaranteed with a slight modification of the first or of the second individual process. Thus, a small change in [~~a process~~] **one of these processes** or the addition of a third process can lead to conflicts and blocking

between the processes which can only be empirically verified. In this context, it is possible that a faulty state of the complete process overcomes an empirical test and thus remains undetected. This is not acceptable, especially with regard to a critical installation with respect to safety since it must be guaranteed in every case that no unpredicted event occurs in the interplay of the processes.

Apart from the unauthorized states to be avoided, there are, in the sequence of a process ~~[so-called authorized states]~~, **"authorized states"** which should occur exclusively for the process if it is functioning correctly.

SUMMARY OF THE INVENTION

It is the object of the invention to specify a method and an arrangement for designing the control of a complete process in which it is (formally) ensured that there is no impediment to the individual processes and **that** only authorized states are occupied.

This object is achieved ~~[in accordance with the features of the independent claims.]~~ **by a method for designing a control of a complete process which comprises a number of individual processes, the method comprising the steps of: identifying functionalities of the individual processes; performing a validation by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, while not impeding each individual process during an operation, producing a validation result; and determining data for controlling the complete process from the validation result.**

~~[Within the context of the invention,]~~ **This object is also achieved by an arrangement for designing the control of a complete process, comprising a number of individual processes; and a processor unit configured to provide: a) identification of functionalities of the individual processes; b) a validation, by automatically verifying an interplay of functionalities in accordance with an input to the complete process, in a manner such that each of the individual processes is not impeded during an operation; and c) data from a result of the validation that is used for controlling the complete process.**

Further developments of the invention include providing a method step

of performing a sequence optimization. A step of producing data for the control in an executable code form may be provided, as may a step of controlling individual affected processes by a software unit which is one of the functionalities of the individual processes. One or more of the individual processes may be an impeding process, an impeding process being defined as such if one of the following conditions is met: an individual process is blocked by another individual process; and an individual process reaches an unauthorized state or a state endangering operation of the complete system. The inventive method may be applied to controlling individual processes of an automatic placement machine, and may also involve controlling a technical installation with data determined for controlling the complete process.

In more detail, the invention relates to a method for designing the control of a complete process which comprises a number of individual processes [~~is specified~~]. In the method, functionalities of the individual processes are identified. Furthermore, a validation is performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, to the effect that each individual process is not disturbed during the operation. From the result of the validation, data for controlling the complete process are determined.

An advantage of the method [~~consists in~~] **is** that [~~it is ensured in~~] the step of validation **ensures** that each individual process can run undisturbed. A further advantage [~~consists in the automatic generation of data~~]**is that data is automatically generated** for controlling the complete process. Thus, data for controlling the complete process are systematically generated with the aid of the method.

An embodiment [~~consists~~] **is provided** in [~~that~~] **which** a sequence optimization is performed after the validation. [~~Thus, it is certainly an advantage that~~] **Advantageously,** individual processes can run undisturbed [~~and another advantage consists in that~~]**; furthermore,** the several individual processes **can** run time-optimized if possible. It is the aim of the sequence optimization to carry out the performance of predetermined actions of the several individual processes in parallel and in the shortest possible time without disturbances.

A further development [~~consists in~~] **is** that the data for controlling the complete

process are determined in the form of an executable code. This ensures that the result of the validation and possibly of the sequence optimization flows completely automatically into the control of the complete process. For example, a program code written in the programming [~~language C or the programming language~~] **languages C or** C++ is generated which initiates or ensures the control of the complete process.

In particular, the advantage becomes noticeable in the generation of executable code if functionalities of the individual processes are also provided in the form of respective program units. If a number of functionalities **in each** of a number of individual processes [~~in each~~] correspond to at least one program unit, the data which were generated in the form of executable codes are used for controlling the coordination of the individual program units or, respectively, the executable code uses the interfaces, e.g.**,** function calls or method calls, provided by the program units.

It is also a further development that an individual process is disturbed if one of the following conditions is met:

a) The individual process is blocked by another individual process. In the case of the blocking, two individual processes wish to use one physical resource in different ways. In such a case, blocking occurs since the resource cannot meet the requirements of both individual processes at the same time.

b) The individual process reaches an unauthorized state or a state endangering the operation of the complete system. It is an essential requirement for a critical application with respect to safety that no hazardous states are assumed.

[~~In the context of the~~] **The** invention[~~,~~] **also provides** an arrangement for controlling a complete process [~~is also specified, which complete process~~] **that** comprises a number of individual processes, a processor unit being provided which is set up in such a manner that functionalities of the individual processes can be identified. Furthermore, a validation can be performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process to the effect that each individual process is not impeded during the operation. Finally, the data resulting from the result of the validation can be used for controlling the complete process.

This arrangement is **particularly** suitable~~[, in particular,]~~ for carrying out the method according to the invention or one of its further developments explained above.

~~[Further developments of the invention are also obtained from the dependent claims.]~~ **BRIEF DESCRIPTION OF THE DRAWINGS**

In the text which follows, exemplary embodiments of the invention are shown and explained with reference to the ~~[drawing, in which:]~~ **drawings.**

~~[Fig. 1 shows]~~ **Fig. 1** **is a schematic diagram showing** a turret head of an automatic placement machine;

Fig. 2 ~~[shows]~~ **is a flowchart showing** steps of a method for generating the control of a complete process;

Fig. 3 ~~[shows]~~ **is** a state ~~[machine which represents]~~ **diagram showing** the system behavior of the "scan" operation;

Fig. 4 ~~[shows]~~ **is** a state ~~[machine which represents]~~ **diagram showing** the specific system behavior of the scan test;

Fig. 5 ~~[shows]~~ **is** a state ~~[machine which represents]~~ **diagram showing** a sequential processing of the vacuum test and of the scan test;

Fig. 6 ~~[shows]~~ **is a state diagram illustrating** two state machines which represent a parallel processing of the vacuum test and of the scan test;

Fig. 7 ~~[shows a number of state machines which represent]~~ **are state diagrams showing** a system behavior;

Fig. 8 ~~[shows]~~ **is** a state ~~[machine which represents]~~ **diagram showing** a specific system behavior **(error recovery);**~~[;]~~

~~[Fig. 9 shows]~~ **Figs. 9A & B** **are** two state ~~[machines]~~ **diagrams** which in each case ~~[represent]~~ **show** specific system behavior for the vacuum test and the scan test;

~~[Fig. 10 shows]~~ **Figs. 10A & B** **are parts of** a state ~~[machine for]~~ **diagram showing** the complete process.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Figure 1 shows **an exemplary embodiment having** a turret head 101 of an

automatic placement machine. The turret head 101 accepts components and places them at a predetermined target position. The turret head contains 12 vacuum pipettes 102 which are used as receptacle and placement tool. If the turret head 101 is used for a prolonged period, wear occurs, **and** the vacuum pipettes become dirty and worn. Accordingly, it is necessary to perform periodic tests in order to determine the state of the vacuum pipettes 102 and to exchange them, if necessary. Two different tests are performed by two different C programs. A vacuum test 103 is used for finding out whether the respective vacuum pipette 102 can still generate the intended vacuum; a scan test 104 indicates the extent to which the individual vacuum pipette 102 is subject to physical wear and whether it needs to be exchanged. For [~~the~~] subsequent observations, the scan test 104 and the [ ]vacuum test 103 access one and the same resource: the rotation of the turret head 101.

The text which follows explains how the control of a complete process is determined, guaranteeing freedom from conflict and providing for the execution of **the** vacuum test 103 and **the** scan test 104 at the same time without the complete process being able to assume unpredicted states. For this purpose, the function calls of the [~~abovementioned~~] **previously mentioned** C programs must be coordinated.

Figure 2 shows steps of a method for generating the control of a complete process.

In a step 201, functionalities of the individual processes are identified (structuring)[~~. Furthermore,~~]**, as well as** controllable and uncontrollable events [~~are identified~~]. Controllable events are events which can be avoided by the control. Uncontrollable events are events which cannot be avoided, e.g.**,** output values of sensors or results of actions. Furthermore, sequences of events are identified which represent a possible physical system behavior. In addition, sequences of events are identified which represent a specific system behavior (task-related system behavior) under the influence of the control.

The step of structuring 201 also comprises the representation of a state machine as shown in figure 3 for the "scan behavior".

From an initial state 301, a "scan" command places the machine into a state 302 in which the vacuum pipette 102 is examined for wear. If the "scanning" is

concluded, the machine returns to state 301. Similarly, the machine returns to [ ]state 301 from state 302 when an error occurs [~~(~~]**(e.g.,** error: the process of scanning indicates that the vacuum pipette 102 must be replaced). A "recover" command changes state 301 to a state 303 in which the machine returns to the starting conditions (recovering). If the "recovering" process is ended, the machine jumps back into state 301 ("done recover").

The specific system behavior is also shown in the form of a state [~~machine~~] **machine/diagram**. For this purpose, [~~figure 4~~] **Figure 4** shows a state machine which corresponds to the specific system behavior for the coordination of the events "turn", "done turn", "error turn", "scan", "done scan", "error scan", **and** "counter".

Figure 4 shows a state machine which represents the specific system behavior of the scan test 104. An initial state 401 is changed to a state 402 by a "turn" command. If the turning of the turret head 101 is ended, the machine changes from state 402 to a state 403. If an error occurs during the turn ("error turn"), state 402 changes to a state 407. From state 403, the "scan" command initiates a change to a state 404; when the scan test 104 is concluded, the machine changes from state 404 to a state 405. Incrementing a counter changes state 405 to a state 406. A check is then made to determine whether the counter has already reached a [~~value 12~~] **particular value, e.g., 12 for a turret heat having 12 pipettes**. If this is so, state 406 is changed to state 407; if the counter exhibits a smaller value than 12, state 406 changes to state 401. Various commands ensure that state 407 is kept: "recover", "done recover", "operator input", **and** "stop". A "repeat" command causes the process to be repeated in that state 407 is changed to state 401.

A next step 202 in figure 2 ensures a validation of the control of the complete process by automatically verifying characteristics of the complete process. Such characteristics are, in particular, a blocking or non-blocking characteristic and a controllability characteristic. If various individual processes are operating in parallel with one another and if these individual processes share one or more resources (in this case the turning of the turret head 101), freedom from blocking is ensured if the individual processes can perform their tasks right to the end without impeding each other by accessing common resources. In the exemplary embodiment shown, the individual process scan test 104 and the individual process vacuum test 103 jointly

use the resource "turning of the turret head 101". This could lead to mutual blocking if the control of the complete process does not [~~prevent this preventatively~~] **avoid this in a preventative manner**.

Furthermore, the validation 202 is carried out in that a plausibility check of the structuring 201 of the complete process to be controlled is effected by observation or simulation of the system and of the specific system behavior in the form of a state machine. Finally, predetermined characteristics are automatically verified. One of these characteristics is [
]"after an error has occurred in scan test 104 (the event "error scan" was indicated), the "recovery" operation (the event "recover") always starts".

The validation 202, if it is not done completely and which formally verifies the undisturbed sequence of the individual processes, is repeated by branching back to step 201, the structuring of the functionalities of the individual processes. If the validation 202 is successful, code for controlling the complete process is automatically generated (compare change to step 203 in [~~figure~~] **Figure** 2).

During this process, during the automatic generation of the control of the complete process, controllable events are allocated, in particular, to the linked function calls within the individual processes and thus to the associated program code fragments. Uncontrollable events are allocated to corresponding return values of function calls or output values of sensors. An example is represented by the function call of the event "scan" which relates to the corresponding program code fragment (C program routine "scan test") which comprises "scan error" or "scan done" as return values.

The automatic generation of the C code for controlling the complete process is determined from various state machines, allocations and/or program code fragments. The individual functionalities structured in step 201 correspond in this case to the corresponding state machines or, respectively, program code fragments.

As already mentioned, the vacuum test 103 and the scan test 104 are carried out in parallel, each test being performed at different physical locations (compare [~~figure~~] **Figure** 1, **noting the** oppositely located performance of the two tests).

Figures 5 and 6 show the desired behavior of the individual processes for the vacuum test 103 and the scan test 104, [~~figure~~] **Figure** 5 showing a sequential

processing of the two tests and ~~[figure]~~ **Figure** 6 showing a parallel processing of the two tests. In the parallel processing in ~~[figure]~~ **Figure** 6, blocking of the two individual processes can occur due to the fact that after the event "recovery vacuum", one of the two events "turn" or "counter" will not occur. As a result, a turning ("turn" command) of the cylinder head, which is needed by both individual processes running in parallel, is not guaranteed. ~~[Whereas one]~~ **One** machine wants to turn the cylinder head, **but** the other machine [
]wants to increment the counter, resulting in blocking. In contrast, sequential processing as indicated in ~~[figure]~~ **Figure** 5 is possible, **but** the tests for 12 vacuum pipettes 102 each being performed in succession ~~[and thus]~~ **results in** the cylinder head 101 having to be turned twice completely. The time expenditure for the sequential processing is far greater than for (almost) parallel processing.

On the basis of ~~[figure]~~ **Figure** 4, ~~[figure]~~ **Figure** 5 to ~~[figure 10]~~ **Figure 10** can be analogously understood ~~[per se. Thus, figure]~~**. Figure** 5 comprises states 501 to 517, figure 6 comprises states 601 to 618, states 501 to 509 and 509 to 517 characterizing ~~[the]~~ in each case **the** independent machines according to ~~[figure]~~ **Figure** 6 which can run in parallel. The event which is in each case decisive for a state changing to another one is in each case indicated along the transition arrows in the figures.

Events having the same name occur synchronously in machines in which the respective event is defined. In the present exemplary embodiment, the event "scan" occurs if the state machine of the system behavior (compare ~~[figure]~~ **Figure** 7) is in state "0" or, respectively, the state machine of the specific system behavior according to ~~[figure]~~ **Figure** 8 is in state 0 or in state 1 and if the state machine of the specific system behavior according to figure 9 is in state 2.

Controllable events are: [
]"turn", "scan", "vacuum", "recover", "recovery turn", "recovery scan", "recovery vacuum", "counter", "operator input", **and** "reset". [
]Uncontrollable events are: [
]"done turn", "error turn", "done scan", "error scan", "done vacuum", "error vacuum", "done recover", "counter=12?", "counter<12?", "stop", **and** "repeat".

The respective state machine indicates the state in which the corresponding

system behavior can be terminated, i.e., the state with a dark background defines a termination condition.

Figure 7 shows ~~[the system behavior. As already mentioned above, the system behavior shows]~~ **as** sequences of events which represent physically possible system states. Such physically possible behaviors are the turning of the cylinder head, the performance of the vacuum test, the performance of the scan test, the incrementing and interrogating of the counter and the inputting of a command which triggers a predetermined action. The state 0 in ~~[figure]~~ **Figure** 7 characterizes both the initial state and the end state of the respective system behavior.

After the system behavior has been identified, the specific system behavior is determined which relates to a behavior of the complete process with regard to the task to be controlled. The associated state machine for the specific system behavior of the error recovery is shown in ~~[figure]~~ **Figure** 8. In ~~[figure]~~ **Figure** 8, there are two marked states 801 and 807, state 801 being simultaneously the start and an end state of the state machine. The specific system behavior "error recovery" can be terminated in each case in states 801 and 807.

According to the above statements, the validation is then performed. To this end, a number of iterations which, finally, lead to the solution according to ~~[figure]~~ **Figure** 7 to ~~[figure]~~ **Figure** 9 are shown according to ~~[figure]~~ **Figure** 2 (compare transition from validation 202 to structuring 201: iteration).

Figures 8 and 9 show the controlled specific system behavior corresponding to the predetermined functionality of the complete process. For this purpose, three tasks have been identified [ ]which are executed in parallel: error recovery, scan test, and vacuum test. The error recovery, in particular, is only activated if both the scan test 104 and the vacuum test 103 ~~[[lacuna]]~~ **occur** in a marked state (compare state 6 ~~[or, respectively]~~ **in Figure 8 or** states 907, 917 in ~~[figure 9)]~~ **Figures 9A & B respectively)**. The scan test 104 and the vacuum test 103 are only activated if the error recovery is in the initial state (compare state 0 or 801 in ~~[figure]~~ **Figure** 8).

The system behavior and the specific system behavior according to ~~[figures]~~ **Figures** 7-9 are non-blocking. Furthermore, the specific system behavior according to ~~[figures]~~ **Figures** 8 and 9 is controllable with respect to the system behavior from

~~[figure 7.]~~ **Figure 7.**

~~[In figure 10]~~ **In Figures 10A & B (collectively, Figure 10)**, the complete process is assembled from the state machines according to ~~[figures]~~ **Figures** 7-9. Figure 10 represents the product state machine of the state machines described ~~[and is comprehensible per se]~~ **above**. In particular, the product state machine according to ~~[figure]~~ **Figure** 10 is not used for structuring and solving the control task for the complete process since the easily traceable procedure, as described, guarantees a structured and clear approach to determining the data which are necessary for controlling the complete process.

The executable program code for controlling the complete process is automatically generated in that first function calls are assigned to the controllable events, the return values of the function calls or output values of sensors being assigned the corresponding uncontrollable events. The program code for controlling the complete process is generated from the state machine assignments and associated program code fragments.

~~[Abstract]~~ **The above-described method and arrangement are illustrative of the principles of the present invention. Numerous modifications and adaptations thereof will be readily apparent to those skilled in this art without departing from the spirit and scope of the present invention.**

~~[Method and arrangement for designing the control of a complete process]~~

ABSTRACT

To prevent blocking between competing individual processes within a complete process, functionalities of the individual processes are identified and the ~~[interplay]~~ **interplay/interaction** of these functionalities is validated. The data obtained from the validation can be used for controlling the complete process.

09/721000
525 Rec'd PCT/PTO 03 JAN 2001

SPECIFICATION

TITLE

METHOD AND ARRANGEMENT FOR DESIGNING THE CONTROL OF A COMPLETE PROCESS

BACKGROUND OF THE INVENTION

Field of the Invention

The invention relates to a method and an arrangement for designing the control of a complete process which comprises a number of individual processes.

Description of the Related Art

The control of a complex technical installation or of a system (a complete process) comprises a number of smaller control units which are provided for certain parts (individual processes) of the installation or of the system. A first control unit for a first individual process is restricted in this case to this individual process. The same applies to a second control for a second individual process. Even if an interplay of the first control with the second control functions largely without errors, this does not guarantee that an error-free operation of the complete installation is still guaranteed with a slight modification of the first or of the second individual process. Thus, a small change in one of these processes or the addition of a third process can lead to conflicts and blocking between the processes which can only be empirically verified. In this context, it is possible that a faulty state of the complete process overcomes an empirical test and thus remains undetected. This is not acceptable, especially with regard to a critical installation with respect to safety since it must be guaranteed in every case that no unpredicted event occurs in the interplay of the processes.

Apart from the unauthorized states to be avoided, there are, in the sequence of a process, "authorized states" which should occur exclusively for the process if it is functioning correctly.

SUMMARY OF THE INVENTION

It is the object of the invention to specify a method and an arrangement for designing the control of a complete process in which it is (formally) ensured that there is no impediment to the individual processes and that only authorized states are occupied.

This object is achieved by a method for designing a control of a complete process which comprises a number of individual processes, the method comprising the steps of: identifying functionalities of the individual processes; performing a validation by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, while not impeding each individual process during an operation, producing a validation result; and determining data for controlling the complete process from the validation result.

This object is also achieved by an arrangement for designing the control of a complete process, comprising a number of individual processes; and a processor unit configured to provide: a) identification of functionalities of the individual processes; b) a validation, by automatically verifying an interplay of functionalities in accordance with an input to the complete process, in a manner such that each of the individual processes is not impeded during an operation; and c) data from a result of the validation that is used for controlling the complete process.

Further developments of the invention include providing a method step of performing a sequence optimization. A step of producing data for the control in an executable code form may be provided, as may a step of controlling individual affected processes by a software unit which is one of the functionalities of the individual processes. One or more of the individual processes may be an impeding process, an impeding process being defined as such if one of the following conditions is met: an individual process is blocked by another individual process; and an individual process reaches an unauthorized state or a state endangering operation of the complete system. The inventive method may be applied to controlling individual processes of an automatic placement machine, and may also

involve controlling a technical installation with data determined for controlling the complete process.

In more detail, the invention relates to a method for designing the control of a complete process which comprises a number of individual processes. In the method, functionalities of the individual processes are identified. Furthermore, a validation is performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, to the effect that each individual process is not disturbed during the operation. From the result of the validation, data for controlling the complete process are determined.

An advantage of the method is that the step of validation ensures that each individual process can run undisturbed. A further advantage is that data is automatically generated for controlling the complete process. Thus, data for controlling the complete process are systematically generated with the aid of the method.

An embodiment is provided in which a sequence optimization is performed after the validation. Advantageously, individual processes can run undisturbed; furthermore, the several individual processes can run time-optimized if possible. It is the aim of the sequence optimization to carry out the performance of predetermined actions of the several individual processes in parallel and in the shortest possible time without disturbances.

A further development is that the data for controlling the complete process are determined in the form of an executable code. This ensures that the result of the validation and possibly of the sequence optimization flows completely automatically into the control of the complete process. For example, a program code written in the programming languages C or C++ is generated which initiates or ensures the control of the complete process.

In particular, the advantage becomes noticeable in the generation of executable code if functionalities of the individual processes are also provided in the form of respective program units. If a number of functionalities in each of a number of individual processes correspond to at least one program unit, the data which were

generated in the form of executable codes are used for controlling the coordination of the individual program units or, respectively, the executable code uses the interfaces, e.g., function calls or method calls, provided by the program units.

It is also a further development that an individual process is disturbed if one of the following conditions is met:

a) The individual process is blocked by another individual process. In the case of the blocking, two individual processes wish to use one physical resource in different ways. In such a case, blocking occurs since the resource cannot meet the requirements of both individual processes at the same time.

b) The individual process reaches an unauthorized state or a state endangering the operation of the complete system. It is an essential requirement for a critical application with respect to safety that no hazardous states are assumed.

The invention also provides an arrangement for controlling a complete process that comprises a number of individual processes, a processor unit being provided which is set up in such a manner that functionalities of the individual processes can be identified. Furthermore, a validation can be performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process to the effect that each individual process is not impeded during the operation. Finally, the data resulting from the result of the validation can be used for controlling the complete process.

This arrangement is particularly suitable for carrying out the method according to the invention or one of its further developments explained above.

BRIEF DESCRIPTION OF THE DRAWINGS

In the text which follows, exemplary embodiments of the invention are shown and explained with reference to the drawings.

Fig. 1 is a schematic diagram showing a turret head of an automatic placement machine;

Fig. 2 is a flowchart showing steps of a method for generating the control of a complete process;

Fig. 3 is a state diagram showing the system behavior of the “scan” operation;

Fig. 4 is a state diagram showing the specific system behavior of the scan test;

Fig. 5 is a state diagram showing a sequential processing of the vacuum test and of the scan test;

Fig. 6 is a state diagram illustrating two state machines which represent a parallel processing of the vacuum test and of the scan test;

Fig. 7 are state diagrams showing a system behavior;

Fig. 8 is a state diagram showing a specific system behavior (error recovery);

Figs. 9A & B are two state diagrams which in each case show specific system behavior for the vacuum test and the scan test;

Figs. 10A & B are parts of a state diagram showing the complete process.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

Figure 1 shows an exemplary embodiment having a turret head 101 of an automatic placement machine. The turret head 101 accepts components and places them at a predetermined target position. The turret head contains 12 vacuum pipettes 102 which are used as receptacle and placement tool. If the turret head 101 is used for a prolonged period, wear occurs, and the vacuum pipettes become dirty and worn. Accordingly, it is necessary to perform periodic tests in order to determine the state of the vacuum pipettes 102 and to exchange them, if necessary. Two different tests are performed by two different C programs. A vacuum test 103 is used for finding out whether the respective vacuum pipette 102 can still generate the intended vacuum; a scan test 104 indicates the extent to which the individual vacuum pipette 102 is subject to physical wear and whether it needs to be exchanged. For subsequent observations, the scan test 104 and the vacuum test 103 access one and the same resource: the rotation of the turret head 101.

The text which follows explains how the control of a complete process is determined, guaranteeing freedom from conflict and providing for the execution of

the vacuum test 103 and the scan test 104 at the same time without the complete process being able to assume unpredicted states. For this purpose, the function calls of the previously mentioned C programs must be coordinated.

Figure 2 shows steps of a method for generating the control of a complete process.

In a step 201, functionalities of the individual processes are identified (structuring), as well as controllable and uncontrollable events. Controllable events are events which can be avoided by the control. Uncontrollable events are events which cannot be avoided, e.g., output values of sensors or results of actions. Furthermore, sequences of events are identified which represent a possible physical system behavior. In addition, sequences of events are identified which represent a specific system behavior (task-related system behavior) under the influence of the control.

The step of structuring 201 also comprises the representation of a state machine as shown in figure 3 for the "scan behavior".

From an initial state 301, a "scan" command places the machine into a state 302 in which the vacuum pipette 102 is examined for wear. If the "scanning" is concluded, the machine returns to state 301. Similarly, the machine returns to state 301 from state 302 when an error occurs (e.g., error: the process of scanning indicates that the vacuum pipette 102 must be replaced). A "recover" command changes state 301 to a state 303 in which the machine returns to the starting conditions (recovering). If the "recovering" process is ended, the machine jumps back into state 301 ("done recover").

The specific system behavior is also shown in the form of a state machine/diagram. For this purpose, Figure 4 shows a state machine which corresponds to the specific system behavior for the coordination of the events "turn", "done turn", "error turn", "scan", "done scan", "error scan", and "counter".

Figure 4 shows a state machine which represents the specific system behavior of the scan test 104. An initial state 401 is changed to a state 402 by a "turn" command. If the turning of the turret head 101 is ended, the machine changes

from state 402 to a state 403. If an error occurs during the turn ("error turn"), state 402 changes to a state 407. From state 403, the "scan" command initiates a change to a state 404; when the scan test 104 is concluded, the machine changes from state 404 to a state 405. Incrementing a counter changes state 405 to a state 406. A check is then made to determine whether the counter has already reached a particular value, e.g., 12 for a turret heat having 12 pipettes. If this is so, state 406 is changed to state 407; if the counter exhibits a smaller value than 12, state 406 changes to state 401. Various commands ensure that state 407 is kept: "recover", "done recover", "operator input", and "stop". A "repeat" command causes the process to be repeated in that state 407 is changed to state 401.

A next step 202 in figure 2 ensures a validation of the control of the complete process by automatically verifying characteristics of the complete process. Such characteristics are, in particular, a blocking or non-blocking characteristic and a controllability characteristic. If various individual processes are operating in parallel with one another and if these individual processes share one or more resources (in this case the turning of the turret head 101), freedom from blocking is ensured if the individual processes can perform their tasks right to the end without impeding each other by accessing common resources. In the exemplary embodiment shown, the individual process scan test 104 and the individual process vacuum test 103 jointly use the resource "turning of the turret head 101". This could lead to mutual blocking if the control of the complete process does not avoid this in a preventative manner.

Furthermore, the validation 202 is carried out in that a plausibility check of the structuring 201 of the complete process to be controlled is effected by observation or simulation of the system and of the specific system behavior in the form of a state machine. Finally, predetermined characteristics are automatically verified. One of these characteristics is "after an error has occurred in scan test 104 (the event "error scan" was indicated), the "recovery" operation (the event "recover") always starts".

The validation 202, if it is not done completely and which formally verifies the undisturbed sequence of the individual processes, is repeated by branching back to step 201, the structuring of the functionalities of the individual processes. If the

validation 202 is successful, code for controlling the complete process is automatically generated (compare change to step 203 in Figure 2).

During this process, during the automatic generation of the control of the complete process, controllable events are allocated, in particular, to the linked function calls within the individual processes and thus to the associated program code fragments. Uncontrollable events are allocated to corresponding return values of function calls or output values of sensors. An example is represented by the function call of the event "scan" which relates to the corresponding program code fragment (C program routine "scan test") which comprises "scan error" or "scan done" as return values.

The automatic generation of the C code for controlling the complete process is determined from various state machines, allocations and/or program code fragments. The individual functionalities structured in step 201 correspond in this case to the corresponding state machines or, respectively, program code fragments.

As already mentioned, the vacuum test 103 and the scan test 104 are carried out in parallel, each test being performed at different physical locations (compare Figure 1, noting the oppositely located performance of the two tests).

Figures 5 and 6 show the desired behavior of the individual processes for the vacuum test 103 and the scan test 104, Figure 5 showing a sequential processing of the two tests and Figure 6 showing a parallel processing of the two tests. In the parallel processing in Figure 6, blocking of the two individual processes can occur due to the fact that after the event "recovery vacuum", one of the two events "turn" or "counter" will not occur. As a result, a turning ("turn" command) of the cylinder head, which is needed by both individual processes running in parallel, is not guaranteed. One machine wants to turn the cylinder head, but the other machine wants to increment the counter, resulting in blocking. In contrast, sequential processing as indicated in Figure 5 is possible, but the tests for 12 vacuum pipettes 102 each being performed in succession results in the cylinder head 101 having to be turned twice completely. The time expenditure for the sequential processing is far greater than for (almost) parallel processing.

On the basis of Figure 4, Figure 5 to Figure 10 can be analogously understood. Figure 5 comprises states 501 to 517, figure 6 comprises states 601 to 618, states 501 to 509 and 509 to 517 characterizing in each case the independent machines according to Figure 6 which can run in parallel. The event which is in each case decisive for a state changing to another one is in each case indicated along the transition arrows in the figures.

Events having the same name occur synchronously in machines in which the respective event is defined. In the present exemplary embodiment, the event "scan" occurs if the state machine of the system behavior (compare Figure 7) is in state "0" or, respectively, the state machine of the specific system behavior according to Figure 8 is in state 0 or in state 1 and if the state machine of the specific system behavior according to figure 9 is in state 2.

Controllable events are: "turn", "scan", "vacuum", "recover", "recovery turn", "recovery scan", "recovery vacuum", "counter", "operator input", and "reset". Uncontrollable events are: "done turn", "error turn", "done scan", "error scan", "done vacuum", "error vacuum", "done recover", "counter=12?", "counter<12?", "stop", and "repeat".

The respective state machine indicates the state in which the corresponding system behavior can be terminated, i.e., the state with a dark background defines a termination condition.

Figure 7 shows as sequences of events which represent physically possible system states. Such physically possible behaviors are the turning of the cylinder head, the performance of the vacuum test, the performance of the scan test, the incrementing and interrogating of the counter and the inputting of a command which triggers a predetermined action. The state 0 in Figure 7 characterizes both the initial state and the end state of the respective system behavior.

After the system behavior has been identified, the specific system behavior is determined which relates to a behavior of the complete process with regard to the task to be controlled. The associated state machine for the specific system behavior of the error recovery is shown in Figure 8. In Figure 8, there are two marked states

801 and 807, state 801 being simultaneously the start and an end state of the state machine. The specific system behavior "error recovery" can be terminated in each case in states 801 and 807.

According to the above statements, the validation is then performed. To this end, a number of iterations which, finally, lead to the solution according to Figure 7 to Figure 9 are shown according to Figure 2 (compare transition from validation 202 to structuring 201: iteration).

Figures 8 and 9 show the controlled specific system behavior corresponding to the predetermined functionality of the complete process. For this purpose, three tasks have been identified which are executed in parallel: error recovery, scan test, and vacuum test. The error recovery, in particular, is only activated if both the scan test 104 and the vacuum test 103 occur in a marked state (compare state 6 in Figure 8 or states 907, 917 in Figures 9A & B respectively). The scan test 104 and the vacuum test 103 are only activated if the error recovery is in the initial state (compare state 0 or 801 in Figure 8).

The system behavior and the specific system behavior according to Figures 7-9 are non-blocking. Furthermore, the specific system behavior according to Figures 8 and 9 is controllable with respect to the system behavior from Figure 7.

In Figures 10A & B (collectively, Figure 10), the complete process is assembled from the state machines according to Figures 7-9. Figure 10 represents the product state machine of the state machines described above. In particular, the product state machine according to Figure 10 is not used for structuring and solving the control task for the complete process since the easily traceable procedure, as described, guarantees a structured and clear approach to determining the data which are necessary for controlling the complete process.

The executable program code for controlling the complete process is automatically generated in that first function calls are assigned to the controllable events, the return values of the function calls or output values of sensors being assigned the corresponding uncontrollable events. The program code for controlling

the complete process is generated from the state machine assignments and associated program code fragments.

The above-described method and arrangement are illustrative of the principles of the present invention. Numerous modifications and adaptations thereof will be readily apparent to those skilled in this art without departing from the spirit and scope of the present invention.

ABSTRACT

To prevent blocking between competing individual processes within a complete process, functionalities of the individual processes are identified and the interplay/interaction of these functionalities is validated. The data obtained from the validation can be used for controlling the complete process.

11/prts

09/721000

GR 98 P 2012

525 Rec'd PCT/PTO 03 JAN 2001

Description

Method and arrangement for designing the control of a complete process

The invention relates to a method and an arrangement for designing the control of a complete process which comprises a number of individual processes.

The control of a complex technical installation or, respectively, of a system (complete process) comprises a number of smaller control units which are provided for certain parts (individual processes) of the installation or, respectively, of the system. A first control unit for a first individual process is restricted in this case to this individual process. The same applies to a second control for a second individual process. Even if an interplay of the first control with the second control functions largely without errors, this does not guarantee that an error-free operation of the complete installation is still guaranteed with a slight modification of the first or of the second individual process. Thus, a small change in a process or the addition of a third process can lead to conflicts and blocking between the processes which can only be empirically verified. In this context, it is possible that a faulty state of the complete process overcomes an empirical test and thus remains undetected. This is not acceptable especially with regard to a critical installation with respect to safety since it must be guaranteed in every case that no unpredicted event occurs in the interplay of the processes.

Apart from the unauthorized states to be avoided, there are in the sequence of a process so-called authorized states which should occur exclusively for the process if it is functioning correctly.

It is the object of the invention to specify a method and an arrangement for designing the control of a complete process in which it is (formally) ensured that there is no impediment to the individual processes and only authorized states are occupied.

This object is achieved in accordance with the features of the independent claims.

Within the context of the invention, a method for designing the control of a complete process which comprises a number of individual processes is specified. In the method, functionalities of the individual processes are identified. Furthermore, a validation is performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, to the effect that each individual process is not disturbed during the operation. From the result of the validation, data for controlling the complete process are determined.

An advantage of the method consists in that it is ensured in the step of validation that each individual process can run undisturbed. A further advantage consists in the automatic generation of data for controlling the complete process. Thus, data for controlling the complete process are systematically generated with the aid of the method.

An embodiment consists in that a sequence optimization is performed after the validation. Thus, it is certainly an advantage that individual processes can run undisturbed and another advantage consists in that the several individual processes run time-optimized if possible. It is the aim of the sequence optimization to carry out the performance of predetermined actions of the several individual processes in parallel and in the shortest possible time without disturbances.

A further development consists in that the data for controlling the complete process are determined in the form of an executable code. This ensures that the result of the validation and possibly of the sequence optimization flows completely automatically into the control of the complete process. For example, a program code written in the programming language C or the programming language C++ is generated which initiates or ensures the control of the complete process.

In particular, the advantage becomes noticeable in the generation of executable code if functionalities of the individual processes are also provided in the form of respective program units. If a number of functionalities of a number of individual processes in each correspond to at least one program unit, the data which were generated in the form of executable codes are used for controlling the coordination of the individual program units or, respectively, the executable code uses the interfaces, e.g. function calls or method calls, provided by the program units.

It is also a further development that an individual process is disturbed if one of the following conditions is met:

a) The individual process is blocked by another individual process. In the case of the blocking, two individual processes wish to use one physical resource in different ways. In such a case, blocking occurs since the resource cannot meet the requirements of both individual processes at the same time.

b) The individual process reaches an unauthorized state or a state endangering the operation of the complete system. It is an essential requirement for a critical application with respect to safety that no hazardous states are assumed.

In the context of the invention, an arrangement for controlling a complete process is also specified, which complete process comprises a number of individual processes, a processor unit being provided which is set up in such a manner that functionalities of the individual processes can be identified. Furthermore, a validation can be performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process to the effect that each individual process is not impeded during the operation. Finally, the data resulting from the result of the validation can be used for controlling the complete process.

This arrangement is suitable, in particular, for carrying out the method according to the invention or one of its further developments explained above.

Further developments of the invention are also obtained from the dependent claims.

In the text which follows, exemplary embodiments of the invention are shown and explained with reference to the drawing, in which:

Fig. 1 shows a turret head of an automatic placement machine;

Fig. 2 shows steps of a method for generating the control of a complete process;

Fig. 3 shows a state machine which represents the system behavior of the "scan" operation;

Fig. 4 shows a state machine which represents the specific system behavior of the scan test;

Fig. 5 shows a state machine which represents a sequential processing of the vacuum test and of the scan test;

Fig. 6 shows two state machines which represent a parallel processing of the vacuum test and of the scan test;

Fig. 7 shows a number of state machines which represent a system behavior;

Fig. 8 shows a state machine which represents a specific system behavior;

Fig. 9 shows two state machines which in each case represent specific system behavior for the vacuum test and the scan test;

Fig. 10 shows a state machine for the complete process.

Figure 1 shows a turret head 101 of an automatic placement machine. The turret head 101 accepts components and places them at a predetermined target position. The turret head contains 12 vacuum pipettes 102 which are used as receptacle and placement tool. If the turret head 101 is used for a prolonged period, wear occurs, the vacuum pipettes become dirty and worn. Accordingly, it is necessary to perform periodic tests in order to determine the state of the vacuum pipettes 102 and to exchange them, if necessary. Two different tests are performed by two different C programs. A vacuum test 103 is used for finding out whether the respective vacuum pipette 102 can still generate the intended vacuum; a scan test 104 indicates the extent to which the individual vacuum pipette 102 is subject to physical wear and whether it needs to be exchanged. For the subsequent observations, the scan test 104 and the

vacuum test 103 access one and the same resource: the rotation of the turret head 101.

The text which follows explains how the control of a complete process is determined, guaranteeing freedom from conflict and providing for the execution of vacuum test 103 and scan test 104 at the same time without the complete process being able to assume unpredicted states. For this purpose, the function calls of the abovementioned C programs must be coordinated.

Figure 2 shows steps of a method for generating the control of a complete process.

In a step 201, functionalities of the individual processes are identified (structuring). Furthermore, controllable and uncontrollable events are identified. Controllable events are events which can be avoided by the control. Uncontrollable events are events which cannot be avoided, e.g. output values of sensors or results of actions. Furthermore, sequences of events are identified which represent a possible physical system behavior. In addition, sequences of events are identified which represent a specific system behavior (task-related system behavior) under the influence of the control.

The step of structuring 201 also comprises the representation of a state machine as shown in figure 3 for the "scan behavior".

From an initial state 301, a "scan" command places the machine into a state 302 in which the vacuum pipette 102 is examined for wear. If the "scanning" is concluded, the machine returns to state 301. Similarly, the machine returns to

state 301 from state 302 when an error occurs (error: the process of scanning indicates that the vacuum pipette 102 must be replaced). A "recover" command changes state 301 to a state 303 in which the machine returns to the starting conditions (recovering). If the "recovering" process is ended, the machine jumps back into state 301 ("done recover").

The specific system behavior is also shown in the form of a state machine. For this purpose, figure 4 shows a state machine which corresponds to the specific system behavior for the coordination of the events "turn", "done turn", "error turn", "scan", "done scan", "error scan", "counter".

Figure 4 shows a state machine which represents the specific system behavior of the scan test 104. An initial state 401 is changed to a state 402 by a "turn" command. If the turning of the turret head 101 is ended, the machine changes from state 402 to a state 403. If an error occurs during the turn ("error turn"), state 402 changes to a state 407. From state 403, the "scan" command initiates a change to a state 404; when the scan test 104 is concluded, the machine changes from state 404 to a state 405. Incrementing a counter changes state 405 to a state 406. A check is then made to determine whether the counter has already reached a value 12. If this is so, state 406 is changed to state 407; if the counter exhibits a smaller value than 12, state 406 changes to state 401. Various commands ensure that state 407 is kept: "recover", "done recover", "operator input", "stop". A "repeat" command causes the process to be repeated in that state 407 is changed to state 401.

A next step 202 in figure 2 ensures a validation of the control of the complete process by automatically verifying characteristics of the complete process. Such characteristics are, in particular, a blocking or non-blocking characteristic and a controllability characteristic. If various individual processes are operating in parallel with one another and if these individual processes share one or more resources (in this case the turning of the turret head 101), freedom from blocking is ensured if the individual processes can perform their tasks right to the end without impeding each other by accessing common resources. In the exemplary embodiment shown, the individual process scan test 104 and the individual process vacuum test 103 jointly use the resource "turning of the turret head 101". This could lead to mutual blocking if the control of the complete process does not prevent this preventatively.

Furthermore, the validation 202 is carried out in that a plausibility check of the structuring 201 of the complete process to be controlled is effected by observation or simulation of the system and of the specific system behavior in the form of a state machine. Finally, predetermined characteristics are automatically verified. One of these characteristics is

"after an error has occurred in scan test 104 (the event "error scan" was indicated), the "recovery" operation (the event "recover") always starts".

The validation 202, if it is not done completely and which formally verifies the undisturbed sequence of the individual processes, is repeated by branching back to step 201, the structuring of the functionalities of the individual processes. If the validation 202 is successful, code for controlling the complete process is automatically generated (compare change to step 203 in figure 2).

During this process, during the automatic generation of the control of the complete process, controllable events are allocated, in particular, to the linked function calls within the individual processes and thus to the associated program code fragments. Uncontrollable events are allocated to corresponding return values of function calls or output values of sensors. An example is represented by the function call of the event "scan" which relates to the corresponding program code fragment (C program routine "scan test") which comprises "scan error" or "scan done" as return values.

The automatic generation of the C code for controlling the complete process is determined from various state machines, allocations and/or program code fragments. The individual functionalities structured in step 201 correspond in this case to the corresponding state machines or, respectively, program code fragments.

As already mentioned, the vacuum test 103 and the scan test 104 are carried out in parallel, each test being performed at different physical locations (compare figure 1, oppositely located performance of the two tests).

Figures 5 and 6 show the desired behavior of the individual processes for the vacuum test 103 and the scan test 104, figure 5 showing a sequential processing of the two tests and figure 6 showing a parallel processing of the two tests. In the parallel processing in figure 6, blocking of the two individual processes can occur due to the fact that after the event "recovery vacuum", one of the two events "turn" or "counter" will not occur. As a result, a turning ("turn" command) of the cylinder head, which is needed by both individual processes running in parallel, is not guaranteed. Whereas one machine wants to turn the cylinder head, the other machine

wants to increment the counter, resulting in blocking. In contrast, sequential processing as indicated in figure 5 is possible, the tests for 12 vacuum pipettes 102 each being performed in succession and thus the cylinder head 101 having to be turned twice completely. The time expenditure for the sequential processing is far greater than for (almost) parallel processing.

On the basis of figure 4, figure 5 to figure 10 can be analogously understood per se. Thus, figure 5 comprises states 501 to 517, figure 6 comprises states 601 to 618, states 501 to 509 and 509 to 517 characterizing the in each case independent machines according to figure 6 which can run in parallel. The event which is in each case decisive for a state changing to another one is in each case indicated along the transition arrows in the figures.

Events having the same name occur synchronously in machines in which the respective event is defined. In the present exemplary embodiment, the event "scan" occurs if the state machine of the system behavior (compare figure 7) is in state "0" or, respectively, the state machine of the specific system behavior according to figure 8 is in state 0 or in state 1 and if the state machine of the specific system behavior according to figure 9 is in state 2.

Controllable events are:

"turn", "scan", "vacuum", "recover", "recovery turn", "recovery scan", "recovery vacuum", "counter", "operator input", "reset".

Uncontrollable events are:

"done turn", "error turn", "done scan", "error scan", "done vacuum", "error vacuum", "done recover", "counter=12?", "counter<12?", "stop", "repeat".

The respective state machine indicates the state in which the corresponding system behavior can be terminated, i.e. the state with a dark background defines a termination condition.

Figure 7 shows the system behavior. As already mentioned above, the system behavior shows sequences of events which represent physically possible system states. Such physically possible behaviors are the turning of the cylinder head, the performance of the vacuum test, the performance of the scan test, the incrementing and interrogating of the counter and the inputting of a command which triggers a predetermined action. The state 0 in figure 7 characterizes both the initial state and the end state of the respective system behavior.

After the system behavior has been identified, the specific system behavior is determined which relates to a behavior of the complete process with regard to the task to be controlled. The associated state machine for the specific system behavior of the error recovery is shown in figure 8. In figure 8, there are two marked states 801 and 807, state 801 being simultaneously the start and an end state of the state machine. The specific system behavior "error recovery" can be terminated in each case in states 801 and 807.

According to the above statements, the validation is then performed. To this end, a number of iterations which, finally, lead to the solution according to figure 7 to figure 9 are shown according to figure 2 (compare transition from validation 202 to structuring 201: iteration).

Figures 8 and 9 show the controlled specific system behavior corresponding to the predetermined functionality of the complete process. For this purpose, three tasks have been identified

which are executed in parallel: error recovery, scan test and vacuum test. The error recovery, in particular, is only activated if both the scan test 104 and the vacuum test 103 [lacuna] in a marked state (compare state 6 or, respectively states 907, 917 in figure 9). The scan test 104 and the vacuum test 103 are only activated if the error recovery is in the initial state (compare state 0 or 801 in figure 8).

The system behavior and the specific system behavior according to figures 7-9 are non-blocking. Furthermore, the specific system behavior according to figures 8 and 9 is controllable with respect to the system behavior from figure 7.

In figure 10, the complete process is assembled from the state machines according to figures 7-9. Figure 10 represents the product state machine of the state machines described and is comprehensible per se. In particular, the product state machine according to figure 10 is not used for structuring and solving the control task for the complete process since the easily traceable procedure, as described, guarantees a structured and clear approach to determining the data which are necessary for controlling the complete process.

The executable program code for controlling the complete process is automatically generated in that first function calls are assigned to the controllable events, the return values of the function calls or output values of sensors being assigned the corresponding uncontrollable events. The program code for controlling the complete process is generated from the state machine assignments and associated program code fragments.

Patent claims

1. A method for designing the control of a complete process which comprises a number of individual processes,

a) in which functionalities of the individual processes are identified,

b) in which a validation is performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, to the effect that each individual process is not impeded during the operation,

c) in which data for controlling the complete process are determined from a result of the validation.

2. The method as claimed in claim 1, in which a sequence optimization is performed in addition to step 1c).

3. The method as claimed in claim 1 or 2, in which the data for the control are determined in the form of an executable code.

4. The method as claimed in one of claims 1 to 3, in which one of the functionalities of the individual processes is a software unit for controlling the individual process affected.

5. The method as claimed in one of the preceding claims, in which an individual process is impeded if one of the following conditions is met:

a) the individual process is blocked by another individual process;

b) the individual process reaches an unauthorized state or a state endangering the operation of the complete system.

6. The method as claimed in one of the preceding claims, in which the control of individual processes of an automatic placement machine is designed.

7. The method as claimed in one of the preceding claims, in which the data determined for controlling the complete process are used for controlling a technical installation.

8. An arrangement for designing the control of a complete process which comprises a number of individual processes, comprising a processor unit which is set up in such a manner that

a) functionalities of the individual processes can be identified;

b) a validation can be performed by automatically verifying an interplay of the functionalities in accordance with an input to the complete process, to the effect that each individual process is not impeded during the operation;

c) data from a result of the validation can be used for controlling the complete process.

GR 98 P 2012

Abstract

Method and arrangement for designing the control of a complete process

To prevent blocking between competing individual processes within a complete process, functionalities of the individual processes are identified and the interplay of these functionalities is validated. The data obtained from the validation can be used for controlling the complete process.

09/721000

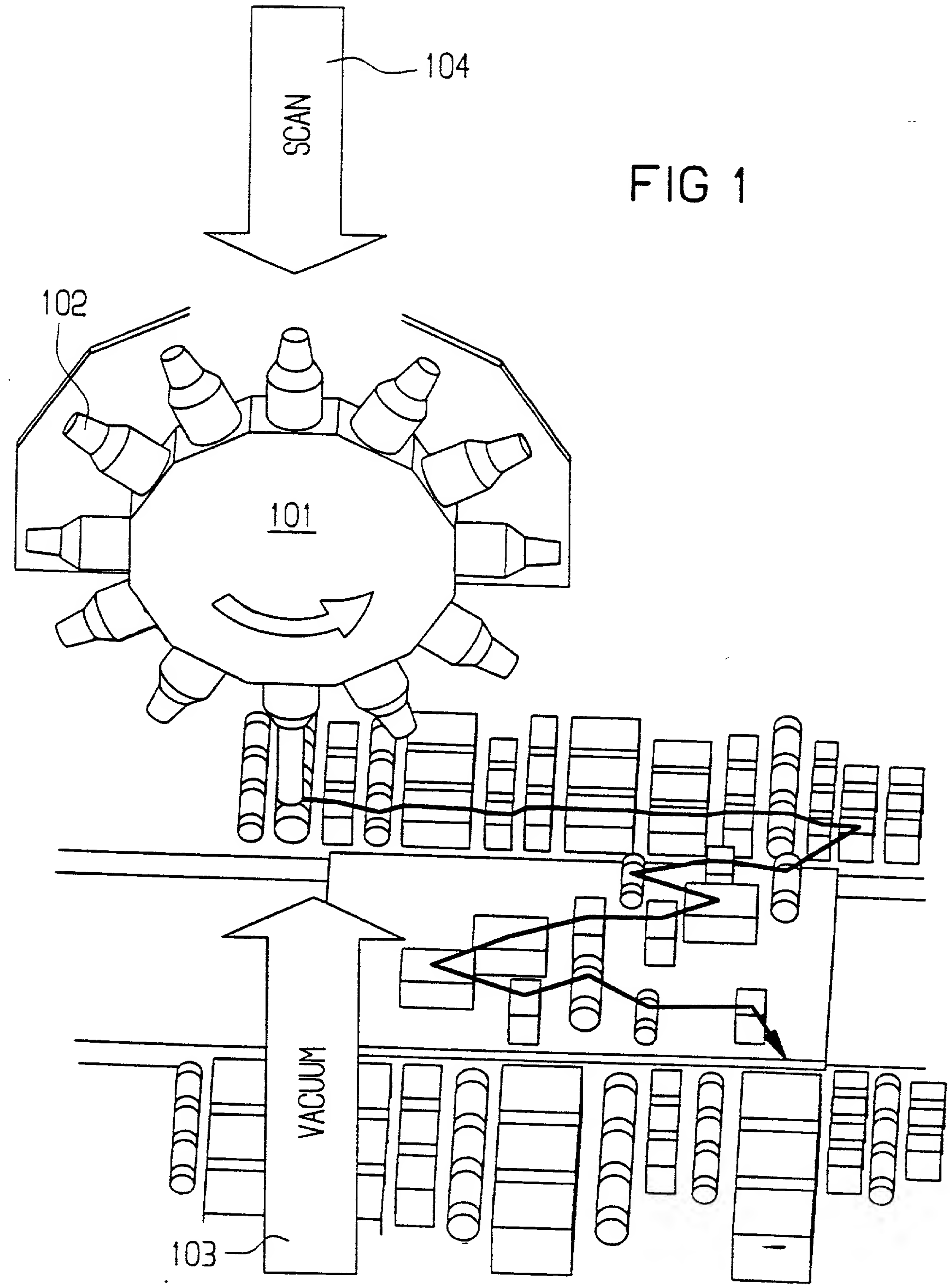


FIG 1

FIG 2

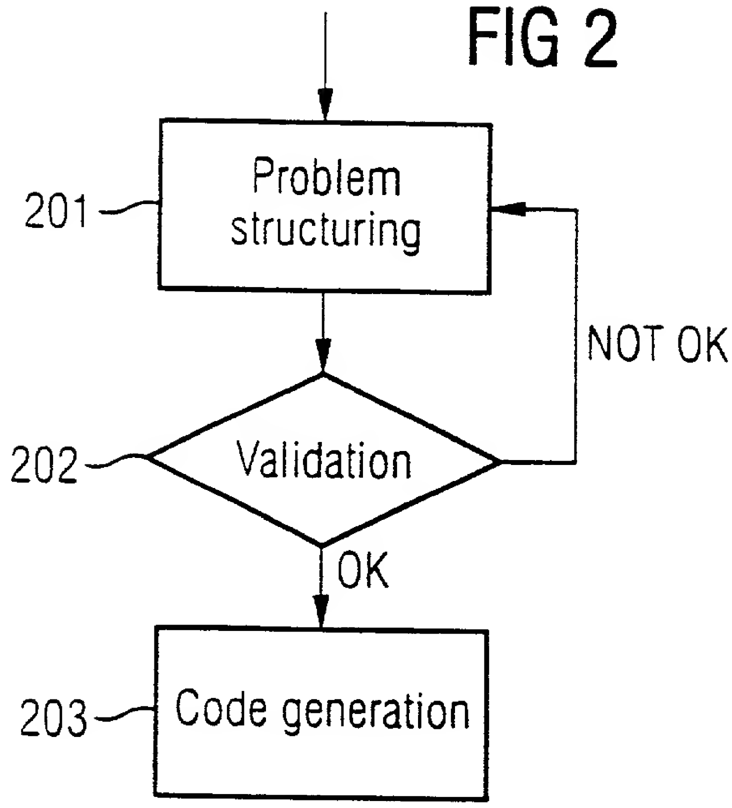

Problem structuring
201
NOT OK
Validation
202
OK
Code generation
203

FIG 3

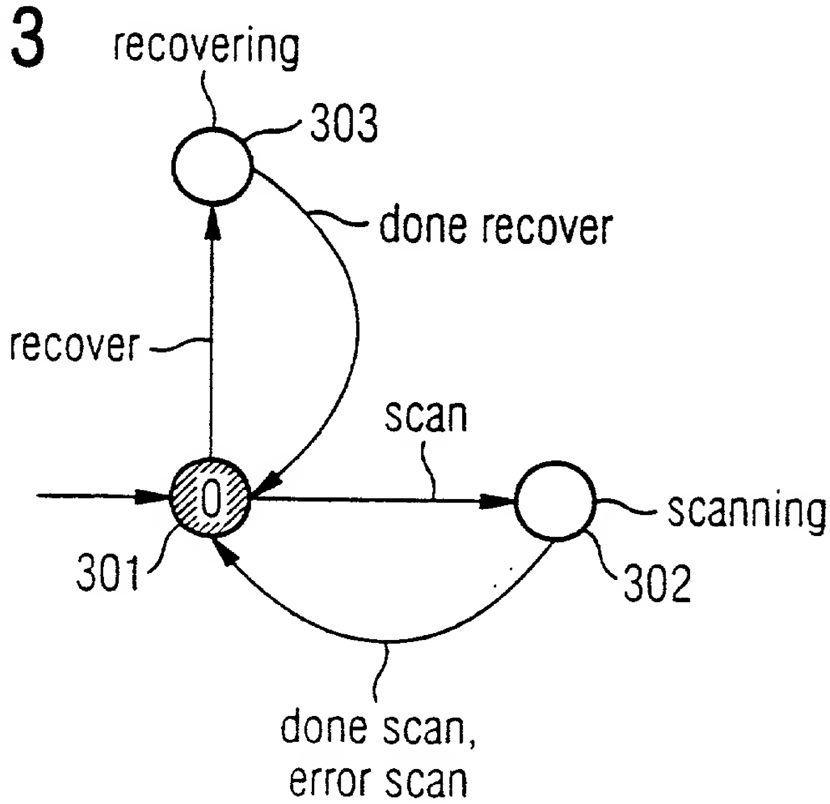
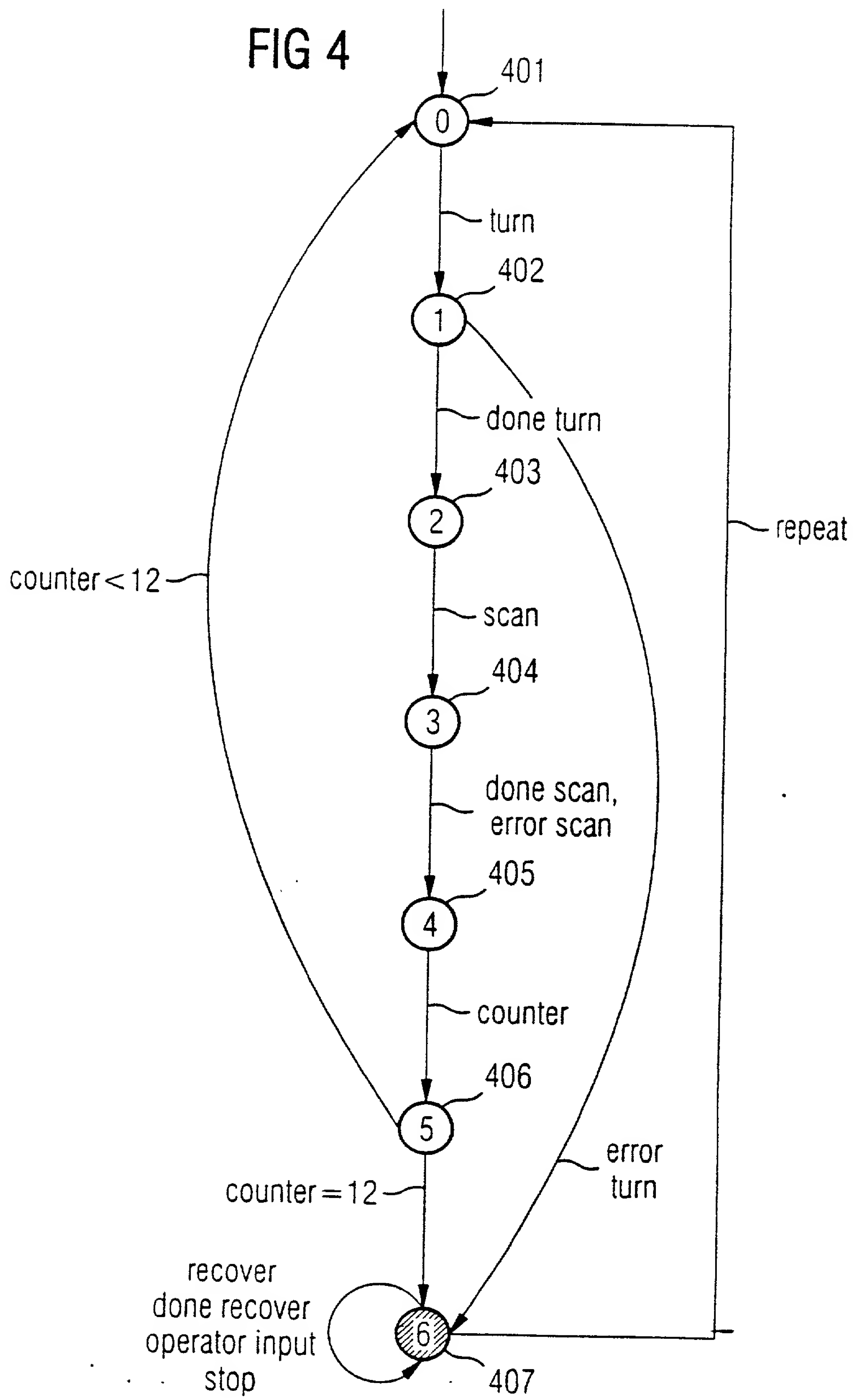
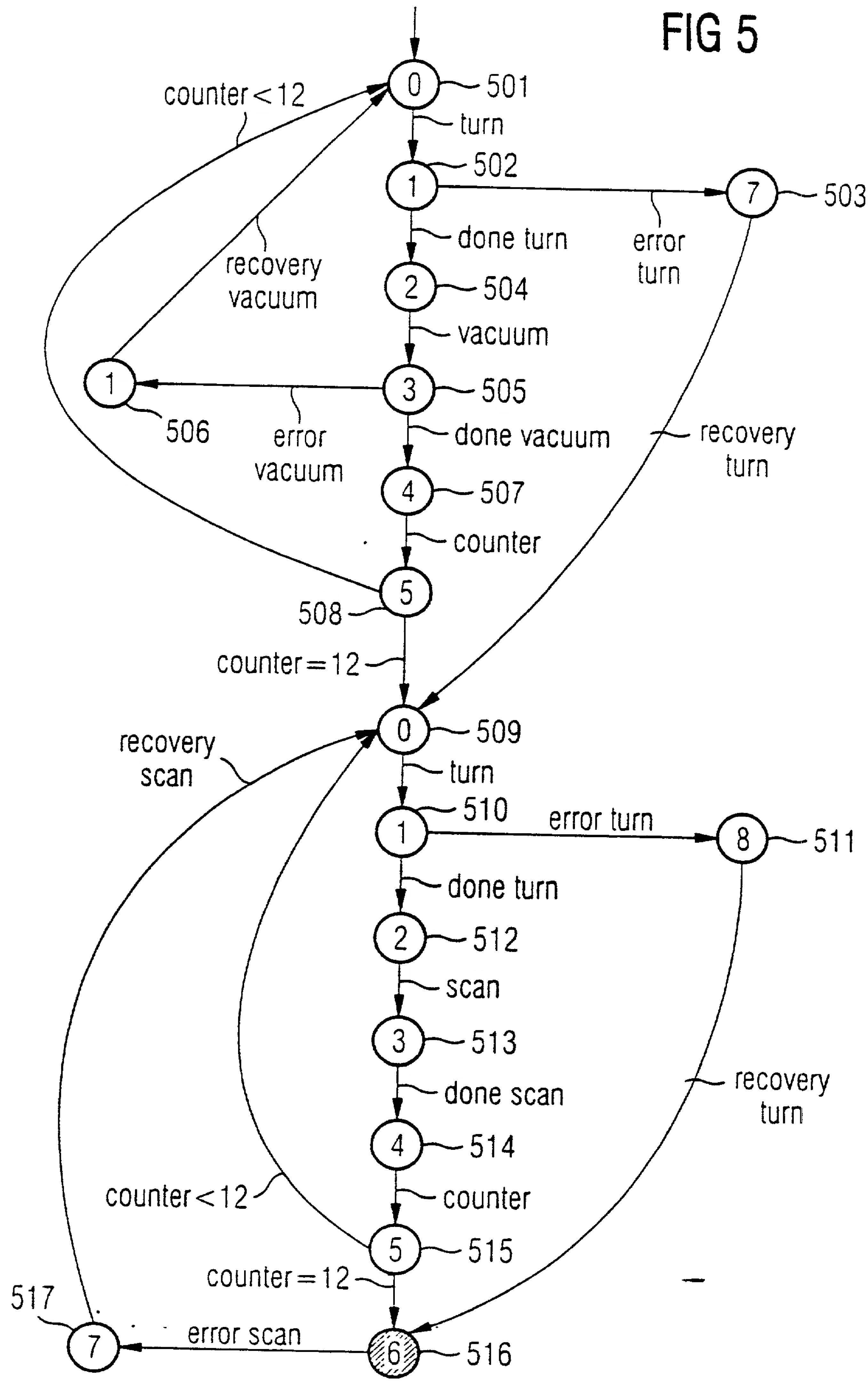

recovering
303
done recover
recover
scan
0
scanning
301
302
done scan,
error scan

FIG 4



401
0
turn
402
1
done turn
403
2
scan
404
3
done scan,
error scan
405
4
counter
406
5
counter<12
counter=12
repeat
error
turn
recover
done recover
operator input
stop
6
407

FIG 5



0
501
turn
1
502
7
503
error turn
done turn
2
504
vacuum
counter<12
recovery vacuum
3
505
1
506
error vacuum
done vacuum
4
507
counter
recovery turn
5
508
counter=12
0
509
recovery scan
turn
1
510
error turn
8
511
done turn
2
512
scan
3
513
done scan
4
514
recovery turn
counter
counter<12
5
515
counter=12
517
7
error scan
6
516

FIG 6

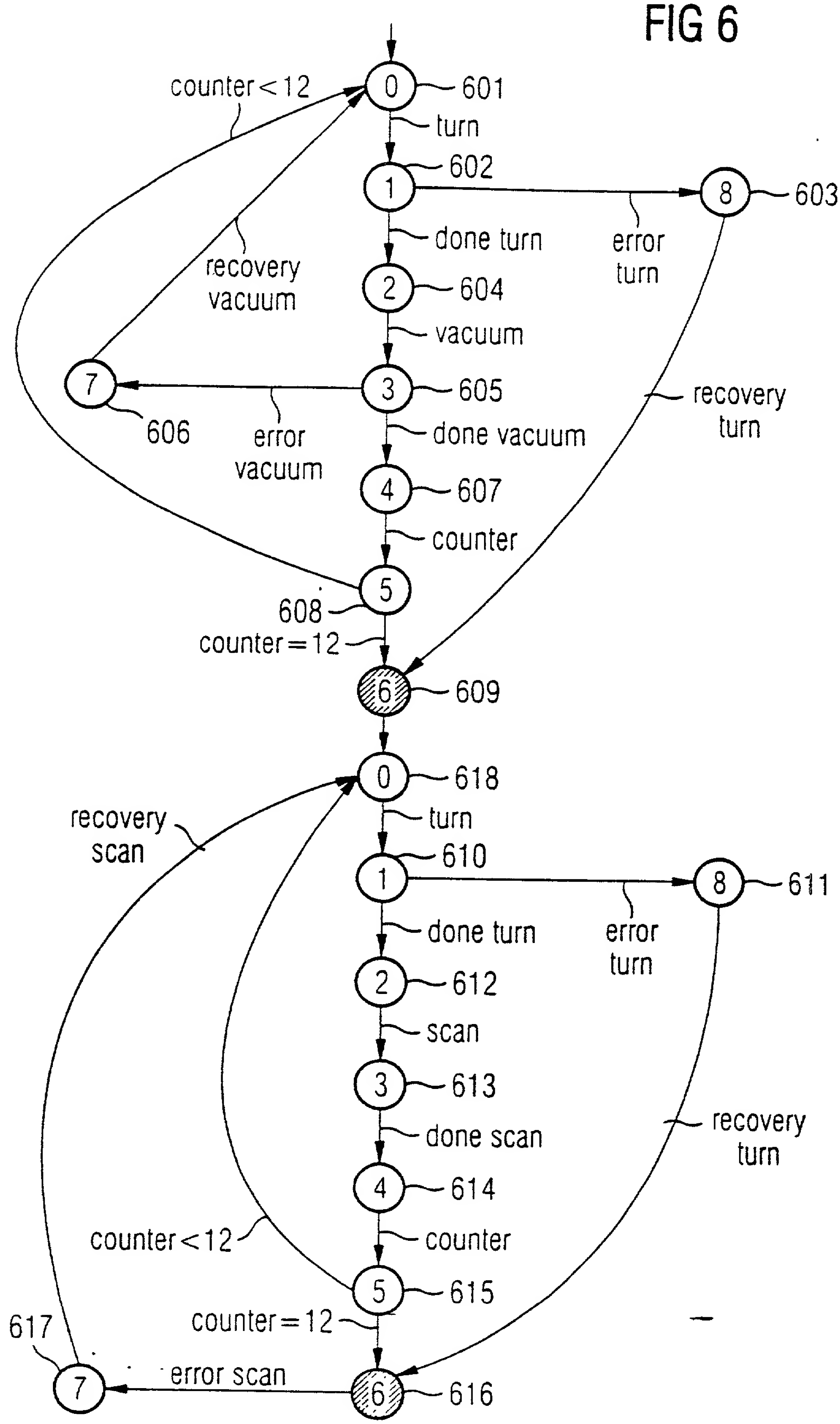

0
601
turn
1
602
8
603
error turn
done turn
2
604
recovery vacuum
counter<12
vacuum
3
605
7
606
error vacuum
done vacuum
recovery turn
4
607
counter
5
608
counter=12
6
609
0
618
turn
recovery scan
1
610
8
611
error turn
done turn
2
612
scan
3
613
done scan
recovery turn
4
614
counter
counter<12
5
615
counter=12
617
7
error scan
6
616

FIG 7

System behaviour

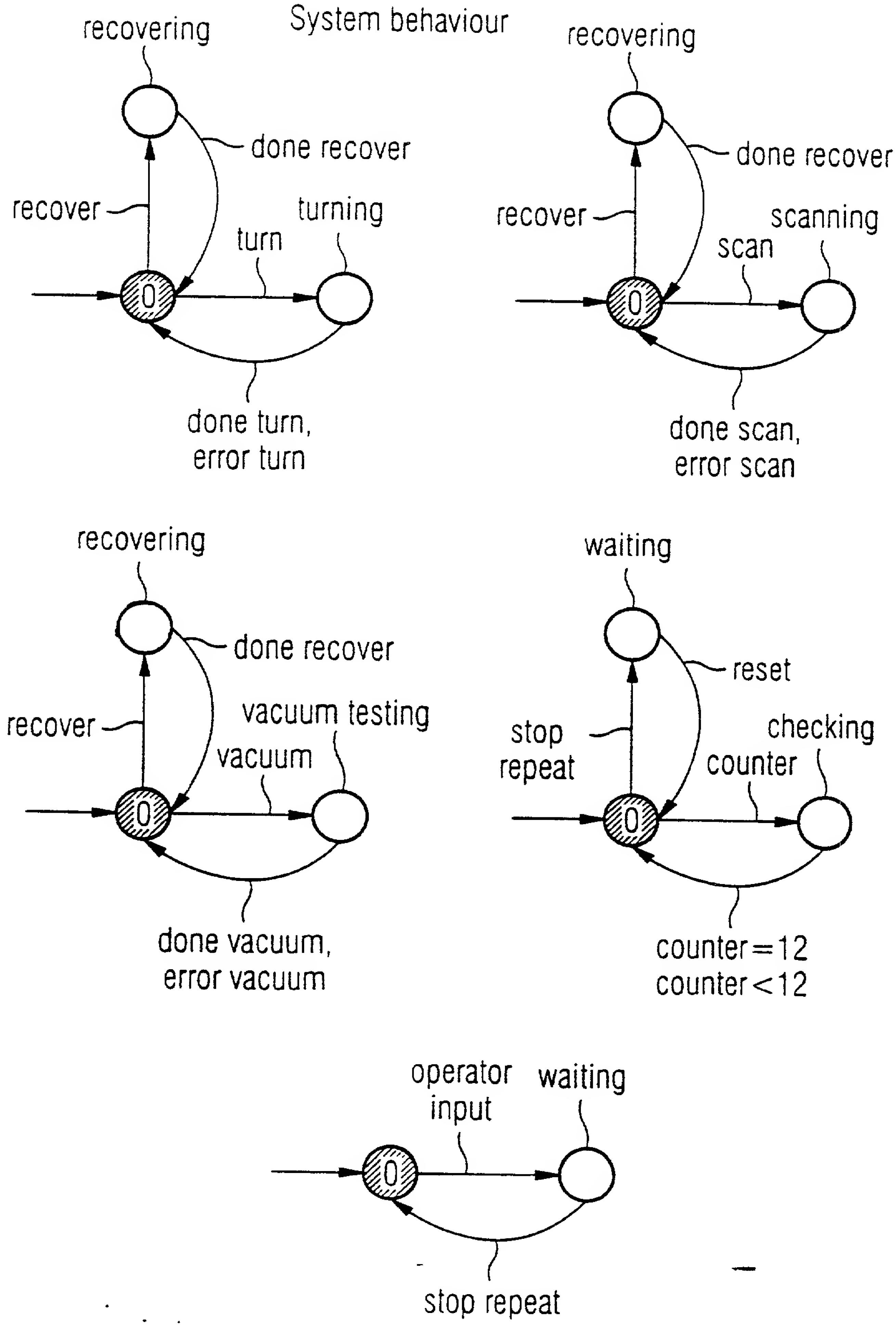


FIG 8 Specified behaviour (error recovery)

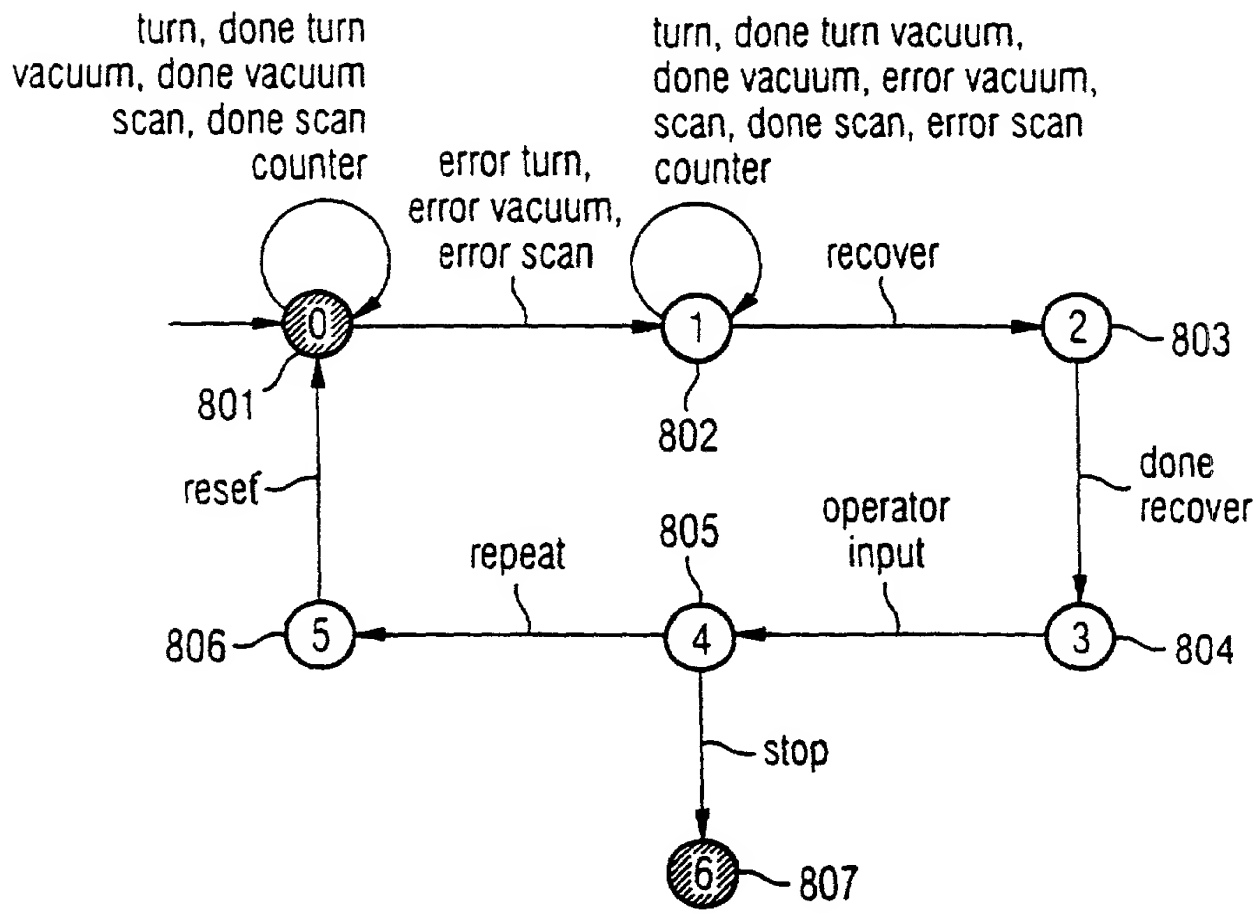
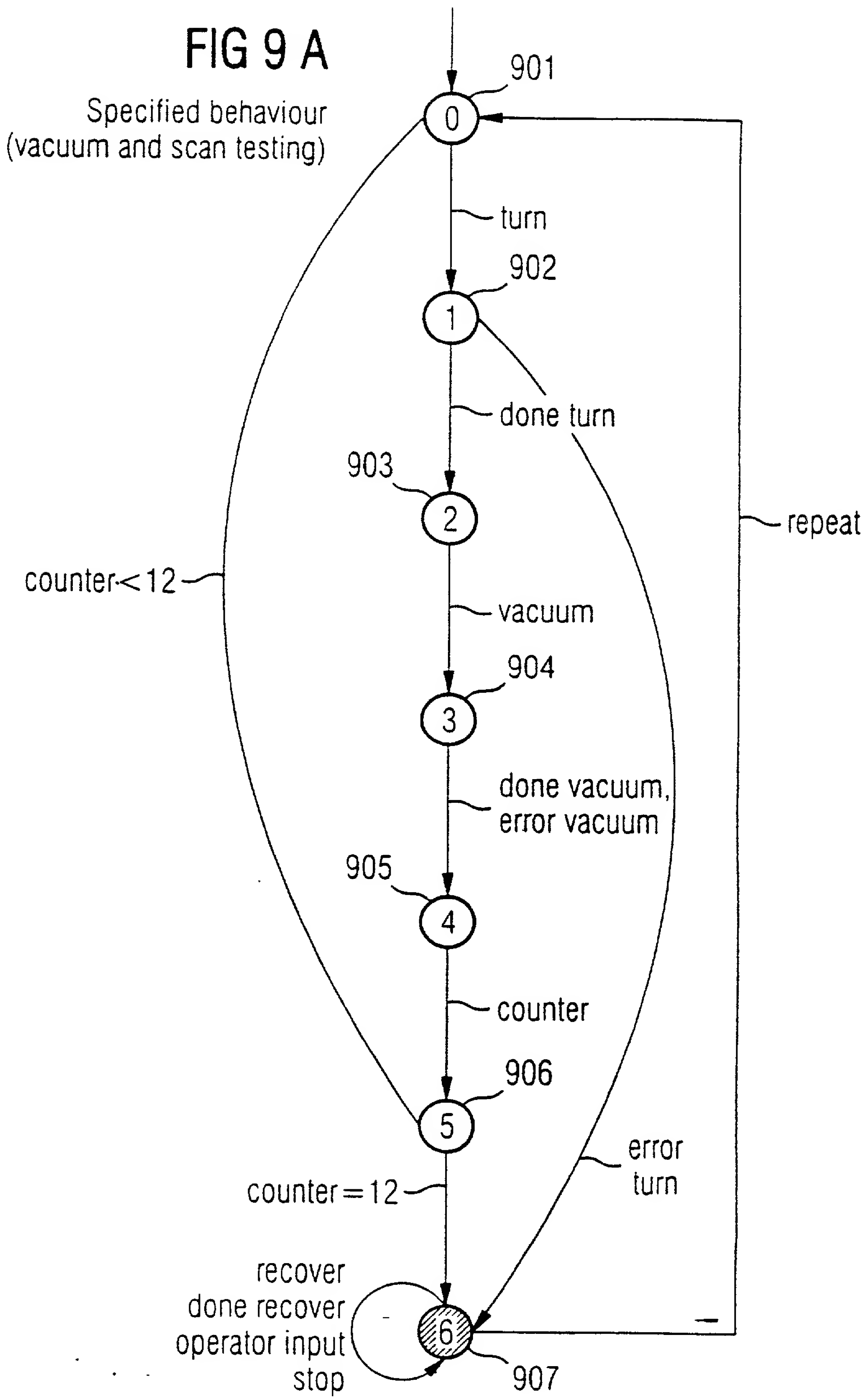


FIG 9 A



901
Specified behaviour
(vacuum and scan testing)
0
turn
902
1
done turn
903
2
vacuum
904
3
done vacuum,
error vacuum
905
4
counter
906
5
counter<12
repeat
error
turn
counter=12
recover
done recover
operator input
stop
6
907

FIG 9 B

Specified behaviour
(vacuum and scan testing)

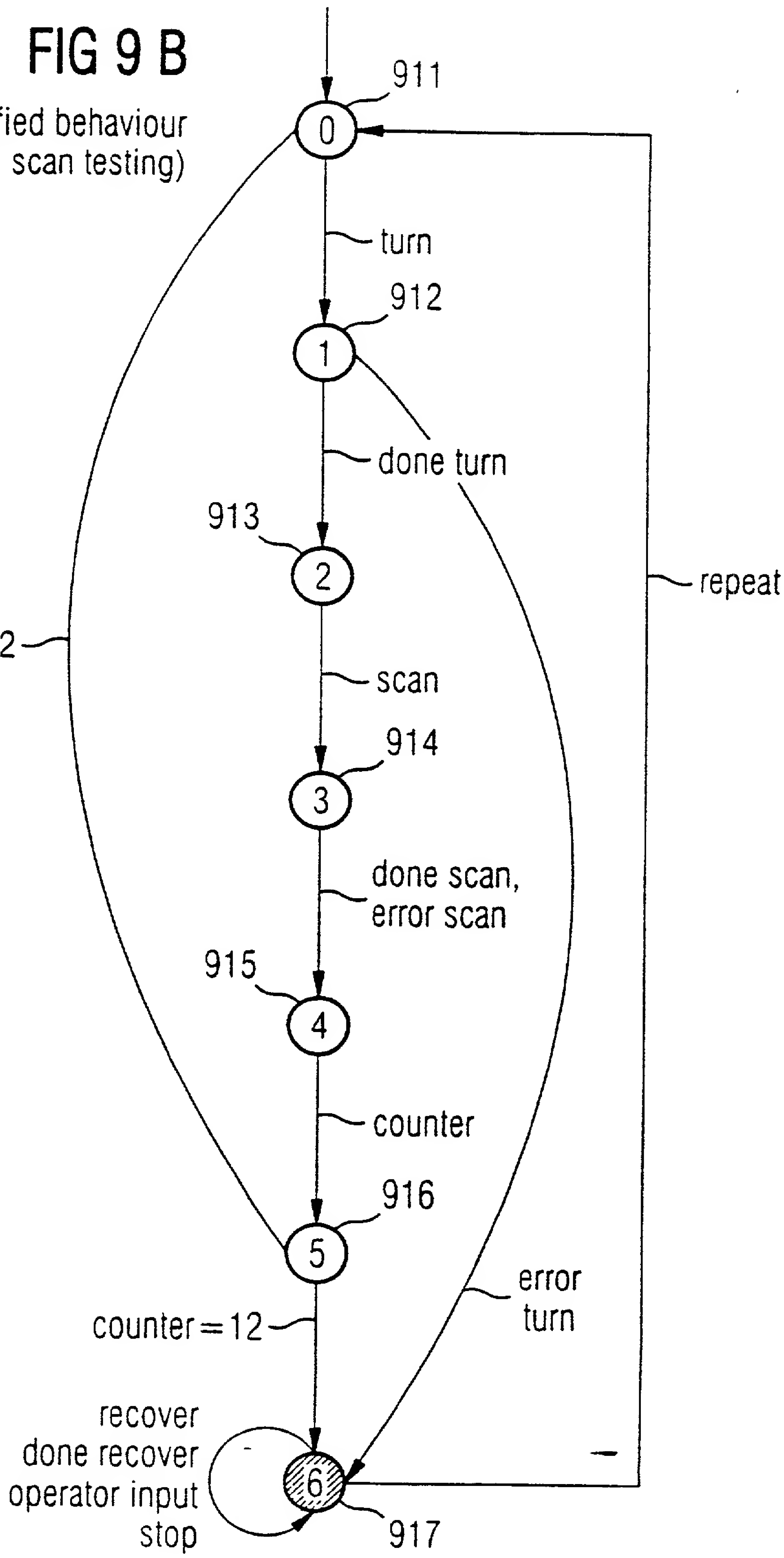
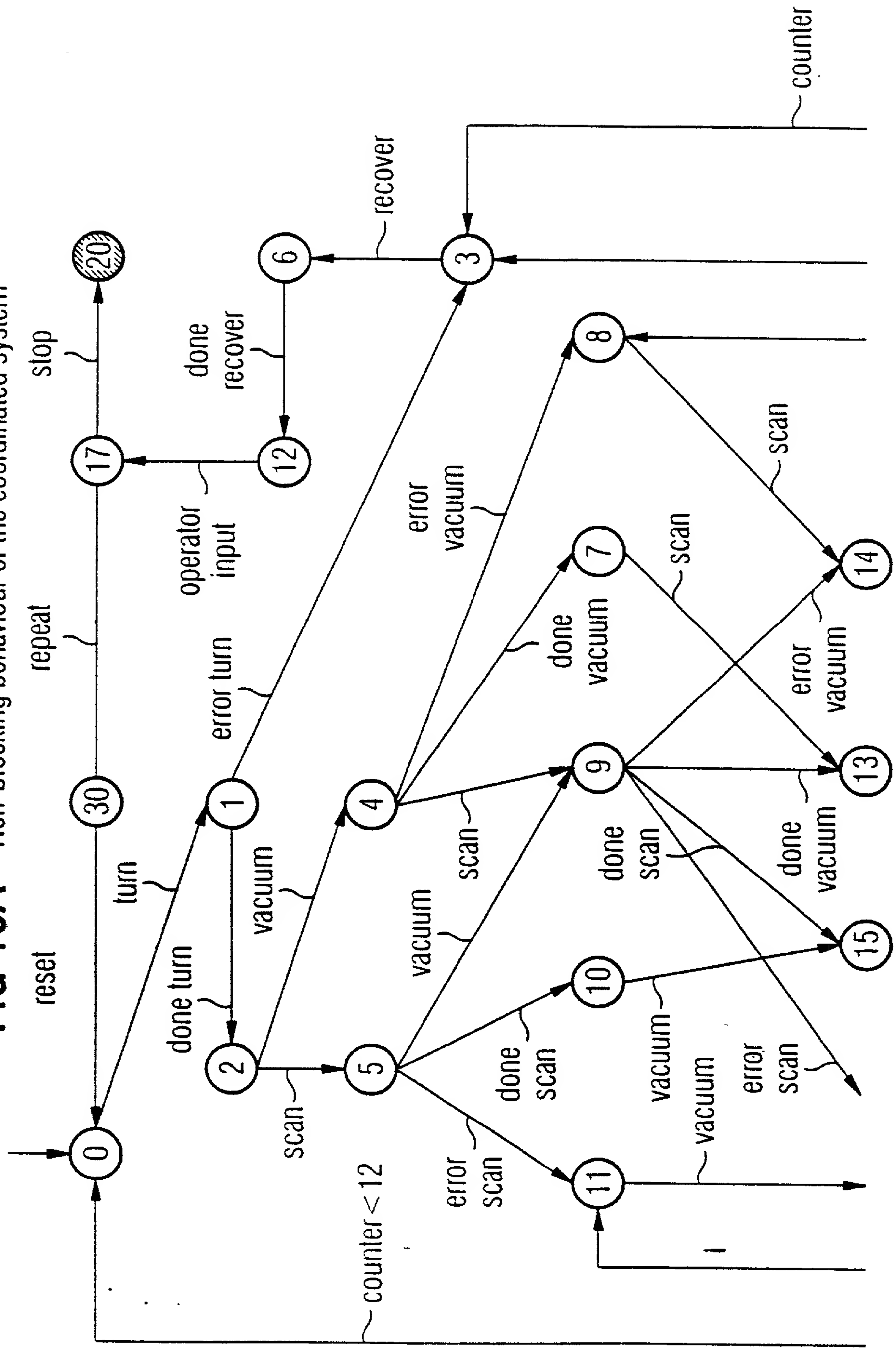
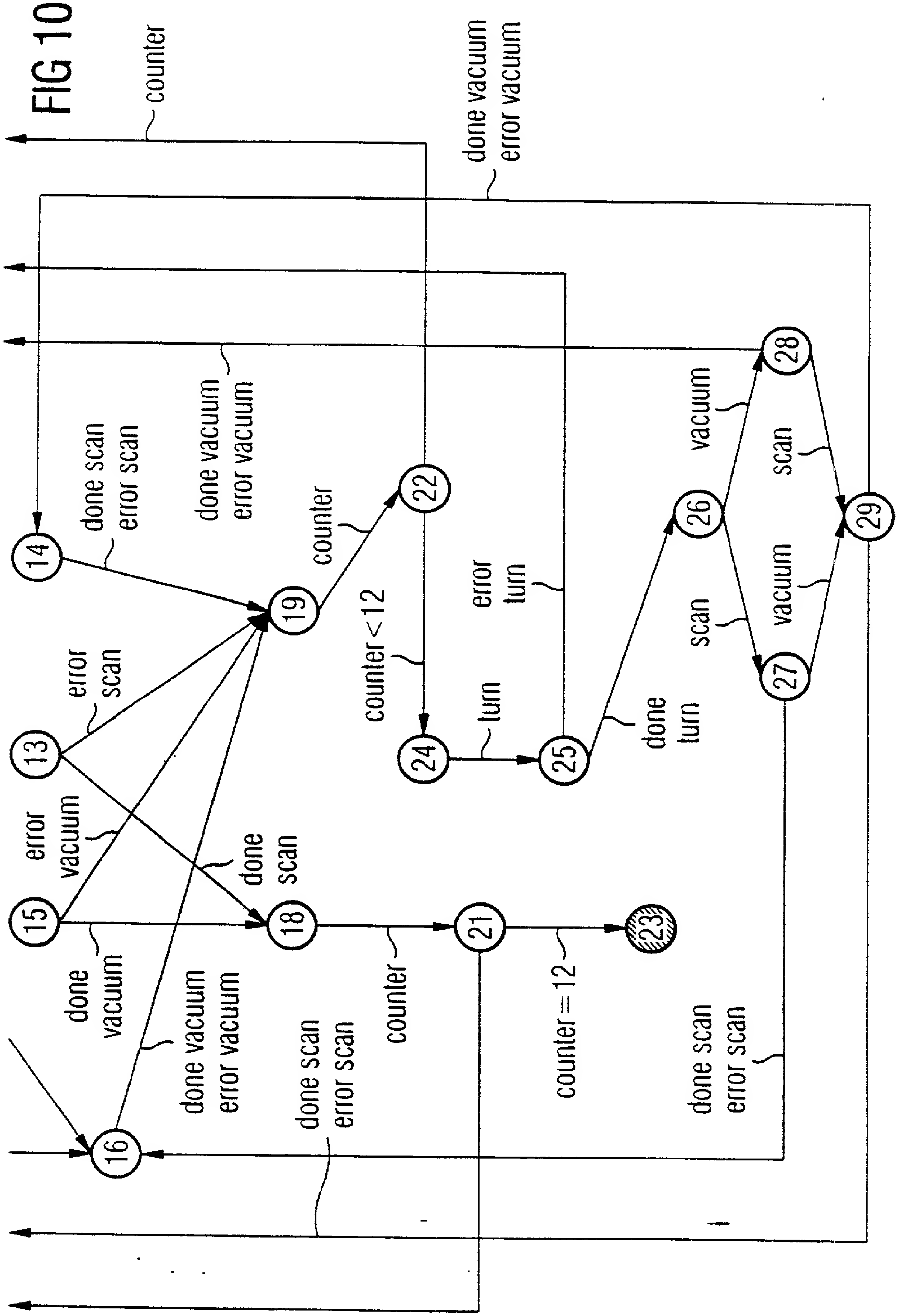


FIG 10A Non-blocking behaviour of the coordinated system



09/721000

FIG 10 B



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CHANGE OF ADDRESS OF APPLICANTS' REPRESENTATIVE

APPLICANT(S):	BERTIL BRANDIN ET AL
ATTORNEY DOCKET NO.:	P00,1982
INTERNATIONAL APPLICATION NO:	PCT/DE99/01915
INTERNATIONAL FILING DATE:	01 JULY 1999
INVENTION:	METHOD AND ARRANGEMENT FOR DESIGNING THE CONTROL OF A COMPLETE PROCESS

Assistant Commissioner for Patents,
Washington, D.C. 20231

S I R:

Members of the firm of Hill & Simpson designated on the original Power of Attorney have merged into the firm of Schiff Hardin & Waite. All future correspondence for the above-referenced application therefore should be sent to the following address:

SCHIFF HARDIN & WAITE
Patent Department
6600 Sears Tower
233 South Wacker Drive
Chicago, Illinois 60606-6473

Submitted by,

(Reg. No. 45,877)

Mark Bergner
SCHIFF HARDIN & WAITE
Patent Department
6600 Sears Tower
Chicago, Illinois 60606-6473
Telephone: (312) 258-5779
Attorneys for Applicants

Declaration and Power of Attorney For Patent Application
Erklärung Für Patentanmeldungen Mit Vollmacht
German Language Declaration

Als nachstehend benannter Erfinder erkläre ich hiermit an Eides Statt:

As a below named inventor, I hereby declare that:

dass mein Wohnsitz, meine Postanschrift, und meine Staatsangehörigkeit den im Nachstehenden nach meinem Namen aufgeführten Angaben entsprechen,

My residence, post office address and citizenship are as stated below next to my name,

dass ich, nach bestem Wissen der ursprüngliche, erste und alleinige Erfinder (falls nachstehend nur ein Name angegeben ist) oder ein ursprünglicher, erster und Miterfinder (falls nachstehend mehrere Namen aufgeführt sind) des Gegenstandes bin, für den dieser Antrag gestellt wird und für den ein Patent beantragt wird für die Erfindung mit dem Titel:

Verfahren und Anordnung zum Entwurf einer Steuerung für einen Gesamtprozeß

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

____

deren Beschreibung

(zutreffendes ankreuzen)

- [x] hier beigefügt ist.
- [ ] am __________ als PCT internationale Anmeldung PCT Anmeldungsnummer __________ eingereicht wurde und am __________ abgeändert wurde (falls tatsächlich abgeändert).

the specification of which

(check one)

- [ ] is attached hereto.
- [ ] was filed on __________ as PCT international application PCT Application No. __________ and was amended on __________ (if applicable)

Ich bestätige hiermit, dass ich den Inhalt der obigen Patentanmeldung einschliesslich der Ansprüche durchgesehen und verstanden habe, die eventuell durch einen Zusatzantrag wie oben erwähnt abgeändert wurde.

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims as amended by any amendment referred to above.

Ich erkenne meine Pflicht zur Offenbarung irgendwelcher Informationen, die für die Prüfung der vorliegenden Anmeldung in Einklang mit Absatz 37, Bundesgesetzbuch, Paragraph 1.56(a) von Wichtigkeit sind, an.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56(a).

Ich beanspruche hiermit ausländische Prioritätsvorteile gemäss Abschnitt 35 der Zivilprozessordnung der Vereinigten Staaten, Paragraph 119 aller unten angegebenen Auslandsanmeldungen für ein Patent oder eine Erfindersurkunde, und habe auch alle Auslandsanmeldungen für ein Patent oder eine Erfindersurkunde nachstehend gekennzeichnet, die ein Anmeldedatum haben, das vor dem Anmeldedatum der Anmeldung liegt, für die Priorität beansprucht wird.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

German Language Declaration

Prior foreign appplications
Priorität beansprucht

Priority Claimed

(Number) (Nummer)	(Country) (Land)	(Day Month Year Filed) (Tag Monat Jahr eingereicht)	Yes Ja	No Nein
198 29 804.8	Germany	03. Juli 1998	☒	☐
			☐	☐
			☐	☐

Ich beanspruche hiermit gemäss Absatz 35 der Zivilprozessordnung der Vereinigten Staaten, Paragraph 120, den Vorzug aller unten aufgeführten Anmeldungen und falls der Gegenstand aus jedem Anspruch dieser Anmeldung nicht in einer früheren amerikanischen Patentanmeldung laut dem ersten Paragraphen des Absatzes 35 der Zivilprozeßordnung der Vereinigten Staaten, Paragraph 122 offenbart ist, erkenne ich gemäss Absatz 37, Bundesgesetzbuch, Paragraph 1.56(a) meine Pflicht zur Offenbarung von Informationen an, die zwischen dem Anmeldedatum der früheren Anmeldung und dem nationalen oder PCT internationalen Anmeldedatum dieser Anmeldung bekannt geworden sind.

I hereby claim the benefit under Title 35. United States Code. §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §122, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a) which occured between the filing date of the prior application and the national or PCT international filing date of this application.

(Application Serial No.) (Anmeldeseriennummer)	(Filing Date) (Anmeldedatum)	(Status) (patentiert, anhängig, aufgegeben)	(Status) (patented, pending, abandoned)

Ich erkläre hiermit, dass alle von mir in der vorliegenden Erklärung gemachten Angaben nach meinem besten Wissen und Gewissen der vollen Wahrheit entsprechen, und dass ich diese eidesstattliche Erklärung in Kenntnis dessen abgebe, dass wissentlich und vorsätzlich falsche Angaben gemäss Paragraph 1001, Absatz 18 der Zivilprozessordnung der Vereinigten Staaten von Amerika mit Geldstrafe belegt und/oder Gefängnis bestraft werden koennen, und dass derartig wissentlich und vorsätzlich falsche Angaben die Gültigkeit der vorliegenden Patentanmeldung oder eines darauf erteilten Patentes gefährden können.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

German Language Declaration

VERTRETUNGSVOLLMACHT: Als benannter Erfinder beauftrage ich hiermit den nachstehend benannten Patentanwalt (oder die nachstehend benannten Patentanwälte) und/oder Patent-Agenten mit der Verfolgung der vorliegenden Patentanmeldung sowie mit der Abwicklung aller damit verbundenen Geschäfte vor dem Patent- und Warenzeichenamt: *(Name und Registrationsnummer anführen)*

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

19

And I hereby appoint

Messrs. John D. Simpson (Registration No. 19,842) Lewis T. Steadman (17,074), William C. Stueber (16,453), P. Phillips Connor (19,259), Dennis A. Gross (24,410), Marvin Moody (16,549), Steven H. Noll (28,982), Brett A. Valiquet (27,841), Thomas I. Ross (29,275), Kevin W. Guynn (29,927), Edward A. Lehmann (22,312), James D. Hobart (24,149), Robert M. Barrett (30,142), James Van Santen (16,584), J. Arthur Gross (13,615), Richard J. Schwarz (13,472) and Melvin A. Robinson (31,870), David R. Metzger (32,919), John R. Garrett (27,888) all members of the firm of Hill, Steadman & Simpson, A Professional Corporation.

Telefongespräche bitte richten an: *(Name und Telefonnummer)*

Direct Telephone Calls to: *(name and telephone number)*

312/876-0200
Ext. ________________

Postanschrift:

Send Correspondence to:

HILL, STEADMAN & SIMPSON
A Professional Corporation
85th Floor Sears Tower, Chicago, Illinois 60606

Voller Name des einzigen oder ursprünglichen Erfinders: BRANDIN, Bertil	Full name of sole or first inventor:
Unterschrift des Erfinders: B. Brandin — Datum: 18-6-99	Inventor's signature — Date
Wohnsitz: D-80469 München, Germany DEX	Residence
Staatsangehörigkeit: Schweden	Citizenship
Postanschrift: Baumstr. 11	Post Office Addess
D-80469 München Bundesrepublik Deutschland	
Voller Name des zweiten Miterfinders (falls zutreffend): KALTENBACH, Markus	Full name of second joint inventor, if any:
Unterschrift des Erfinders: M. Kaltenbach — Datum: 21.6.1999	Second Inventor's signature — Date
Wohnsitz: D-80801 München, Germany DEX	Residence
Staatsangehörigkeit: Bundesrepublik Deutschland	Citizenship
Postanschrift: Konradstr. 10A	Post Office Address
D-80801 München Bundesrepublik Deutschland	

(Bitte entsprechende Informationen und Unterschriften im Falle von dritten und weiteren Miterfindern angeben).

(Supply similar information and signature for third and subsequent joint inventors).

1-00

2-00